AMSTRAD MAGAZINE

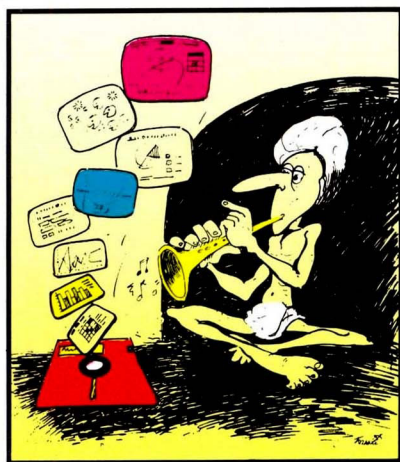
REVISTA DOS UTILIZADORES AMSTRAD

REVISTA MENSAL Nº 5 ANO 1 SETEMBRO 1988 350$00

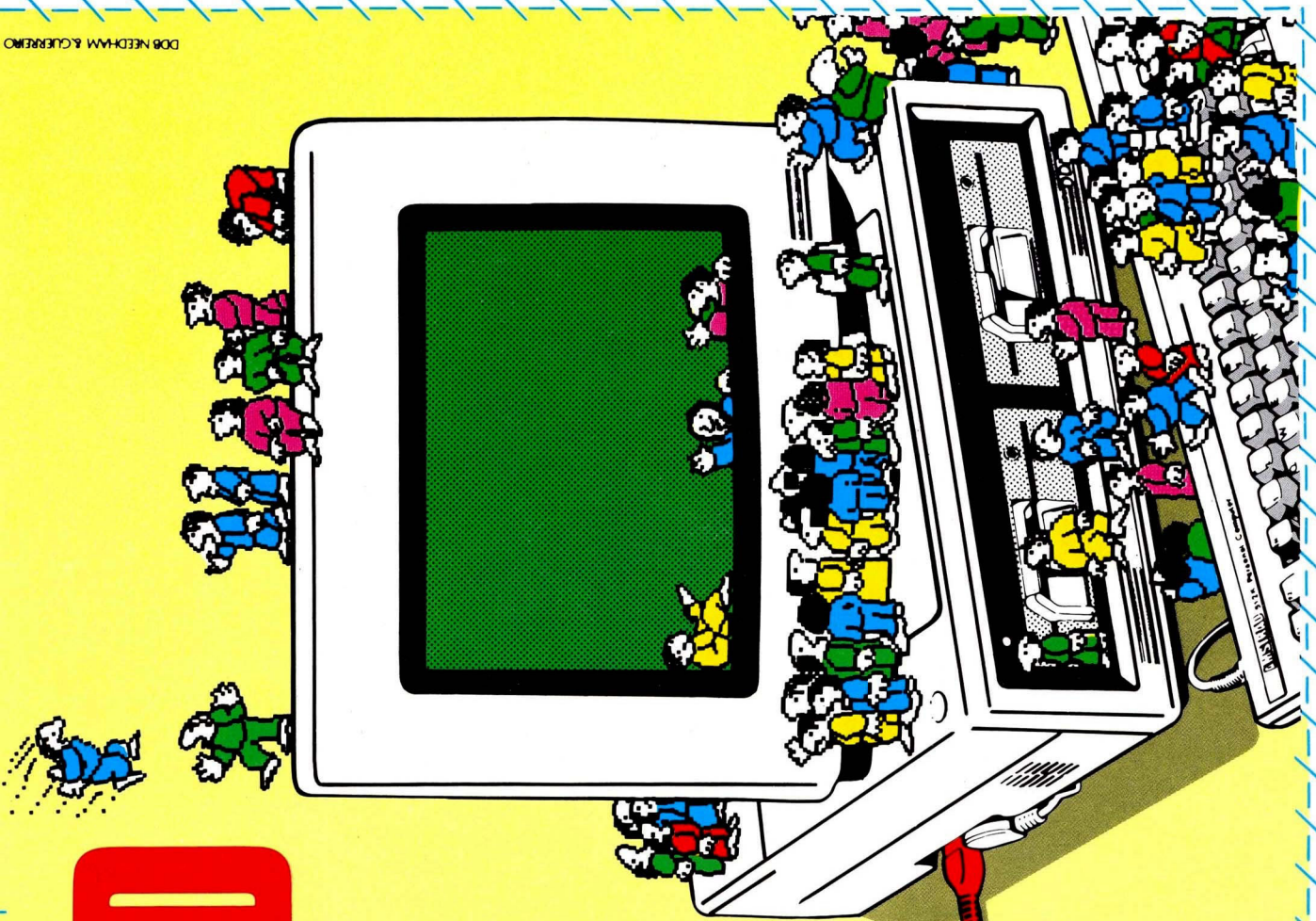
AMSTRAD PS/2?

LINGUAGENS DE PROGRAMAÇÃO

COMO UTILIZAR O DEBUG

ASSEMBLER 8086 É TÃO SIMPLES COMO..

FREE-SOFT: MUITOS MEGAS POR POUCOS ESCUDOS

AMSTRAD
Queremos que a informática chegue a todos.
Por isso facilitámos as coisas... tornando acessível
o que parecia inacessível.
Bastou que a qualidade AMSTRAD, custasse
realmente o que ela custa: o seu preço real –
nem mais nem menos. Assim, tão simples! Tão simples
como utilizar um dos nossos computadores AMSTRAD.
O que os outros complicaram... nós
simplificámos. Porque queremos realmente
que a informática chegue a todos.
Visite um revendedor autorizado
AMSTRAD.
Cominfor
COMPANHIA PORTUGUESA DE INFORMÁTICA
AMSTRAD
Praça de Alvalade, 2-E Esq. – 1700 LISBOA – Telef. (01) 76 90 24
Av. da Boavista, 2881 – Loja 12 – 4100 PORTO – Telef. (02) 68 99 88
DOB NEEDHAM & GUERREIRO

EDITORIAL

REVISTA MENSAL
Nº 5 ANO 1
SETEMBRO 1988
PREÇO 350$00

PROPRIEDADE:
PUBLINFOR, Publicações e Comércio de Artigos de Informática, S.A. — Centro de Escritórios das Laranjeiras — Urbanização das Laranjeiras — Praça Nuno Rodrigues dos Santos, 7-2º Piso - Sala 13 - 1600 LISBOA
Telf: 7269011 Telex 62752 Simose P
Fax: 7269985

DIRECÇÃO:
Fernando Prata

COLABORADORES:
Eng. Mário Leite,
Dr. Maria de Lurdes Leite, António Torres Martins, António Cardoso

PRODUÇÃO GRÁFICA:
SOCEDITE, Lda. — Av. da República, 47-1º Dtº 1000 LISBOA Tels: 767326/767339/768911/760809
Telex: 65016 CEBRO P Fax:732056

PUBLICIDADE:
SOCEDITE, Lda.
— Rua Alfredo Roque Gameiro, 21 - 1º Dtº — Telfs: 762732/767326/767339

ASSINATURAS:
PUBLINFOR

TIRAGEM: 11500 exemplares

PREÇO DE CAPA: 350$00

DISTRIBUIÇÃO:
ELECTROLIBER

— Nº PES. COLECT. 970657668 —
Nº REG. D.G.C.S. 112959
— DEPÓSITO LEGAL
Nº 20669/88

Linguagens de programação: uma dificil opção

Umas mais dificeis, outras mais fáceis, elas estão presentes para satisfazer as necessidades de qualquer utilizador, pouco preocupado em conhecer a máquina com que trabalha, mas desejoso de concluir tarefas que o incansável computador pode levar rápidamente a bom termo. Falamos, evidentemente, das linguagens de programação.

Tal como em qualquer outro dominio da actividade humana, as opiniões acerca do que é, ou não é, uma boa linguagem de programação, na maior parte dos casos divergem de pessoa para pessoa, levando os utilizadores mais sensatos a defenderem a existência de diversas linguagens de programação excelentes (dependendo esta qualidade da aplicação que nelas se pretende desenvolver), e alguns dos utilizadores mais ousados a afirmarem que a linguagem X (que por acaso até é a que dominam melhor) é a melhor linguagem de programação até hoje desenvolvida. Entre todos os utilizadores com posição definida nesta "discussão regular" existe, no entanto, uma enorme percentagem de pessoas com pouco conhecimento das caracteristicas fundamentais das diferentes linguagens de programação, muitas vezes "agarradas" a um dialecto de BASIC pouco standard, que agora começam a sentir a necessidade de fugir para uma linguagem que lhes permita conceber aplicações especificas, de uma forma fácil. Neste número, procurámos facilitar a vida a todos os leitores nessas condições, proporcionando-lhes não só uma conversa com Pedro Veiga, um dos especialistas portugueses na "arte de bem programar", mas também a primeira parte de um dossier sobre a matéria em causa, e ainda um pequeno conjunto de artigos destinados a melhor esclarecer alguns dos aspectos da programação em Assembler 8086/8088.

Tudo isto não é suficiente - nós sabemos - para quem espera durante cerca de 30 dias pela revista que lhes proporciona alguns conhecimentos, algumas ajudas, e mesmo algum gozo diante do teclado. Por isso mesmo nas restantes páginas se podem encontrar algumas das secções já habituais com temas de interesse para os utilizadores de PC's, CPC's, e/ou PCW's, tentando, pelo menos, manter a qualidade que todos desejamos na AM.

O destaque do mês poderá situar-se entre a noticia do PS/2 Amstrad, que constitui o nosso artigo de capa, e o lançamento de uma nova secção criada com o fim de possibilitar aos leitores a aquisição de boas "peças de software", a preços super-reduzidos. Mas, falando de novidades, é bom referir que não ficámos por aqui. Os mais observadores concerteza já notaram o aumento de páginas (sem aumento de custo!!!) e algumas alterações de formato de determinadas secções. A justificação deve-se ao desejo de apresentar em cada mês uma revista melhor, quer sob o aspecto gráfico, quer sob o aspecto de conteúdo.

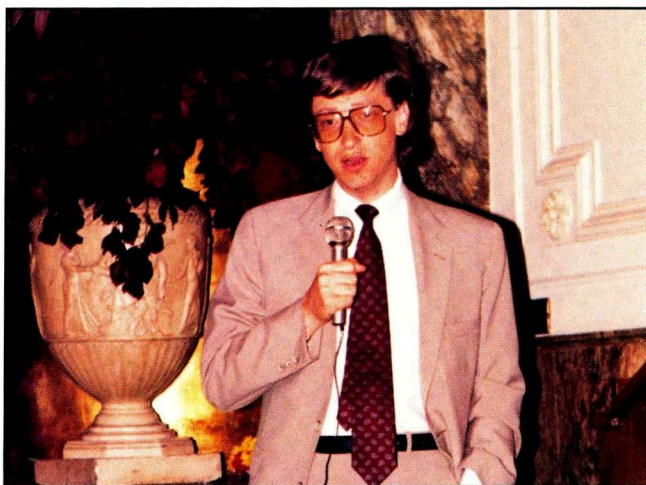
Neste processo de melhoramento gradual, relembramos por fim que continuamos á espera da vossa ajuda através de sugestões e trabalhos que, mesmo pequenos, serão sempre acolhidos com agrado.

SUMÁRIO

BILL GATES esteve em Espanha

"PC's E PS's VÃO COEXISTIR"

Decorreram poucos dias desde a visita de Bill Gates, presidente da Microsoft, a Espanha. A AMSTRAD MAGAZINE foi convidada a estar presente e traz aos seus leitores as perspectivas daquele que se pode considerar o líder do software internacional.

A Microsoft foi a empresa que concebeu o MS-DOS e o MSX-DOS, os sistemas operativos para microcomputadores de 16 e 8 bits, respectivamente. Recentemente, foi ainda a Microsoft que liderou, sob encomenda da IBM, o desenvolvimento do OS/2.

Interrogado sobre a concorrência entre PS's e PC's, Bill Gates escusou-se a responder sobre questões relacionadas com a estratégia da IBM. Mas, por outro lado, não hesitou em afirmar que os standards PS e PC vão coexistir num futuro longo. Como prova disso, Bill Gates anunciou que em breve ficarão disponíveis novas versões do MS-DOS e, obviamente, "a Microsoft não aposta em perspectivas sem futuro".

100 000 ANTENAS POR MÊS AMSTRAD — FIDELITY

No acordo celebrado entre a AMSTRAD e o conhecido magnata da imprensa e televisão RUPERT MURDOCK, a companhia de Alan Sugar comprometeu-se a fabricar antenas parabólicas adaptadas às emissões retransmitidas pelo satélite ASTRA, ao ritmo de 100 000 unidades por mês, durante o ano de 1989.

Está efectivamente aberta uma nova era no que respeita à televisão via satélite. O ASTRA é o primeiro satélite totalmente vocacionado à retransmissão de programas de televisão, ao contrário dos actuais que dividem as suas actividades entre as comunicações e a TV. Por outro lado, a tecnologia actualizada reporta-o a um satélite de média potência ao invés do que actualmente conhecemos, satélites de baixa potência.

Tudo isto conduz a que dos "muito ricos", que montam os seus sistemas de recepção TV via satélite por mais de cinco centenas de milhares de escudos ou os exemplos de "fraternais comunidades" que conseguiram para seu condomínio inteiro os mesmos efeitos por mais ou menos 2 000 contos deixem de ser previligiados. O novo satélite e os novos sistemas vão permitir que mesmo em Portugal, cada um possa adquirir a sua própria "frigideira" (antena parabólica) por menos de cem contos.

Isto é, sem se "chatear" com os condomínios ou com a administração do condomínio, ou sem fazer investimentos exagerados, cada um poderá adquirir o seu sistema de recepção TV, via satélite, por um preço similar ao de um vídeo.

Para já, não tem que saber línguas. O novo sistema, aceita a imagem e o som separadamente, o que quer dizer que se pode sintonizar, para uma mesma imagem a língua que se desejar.

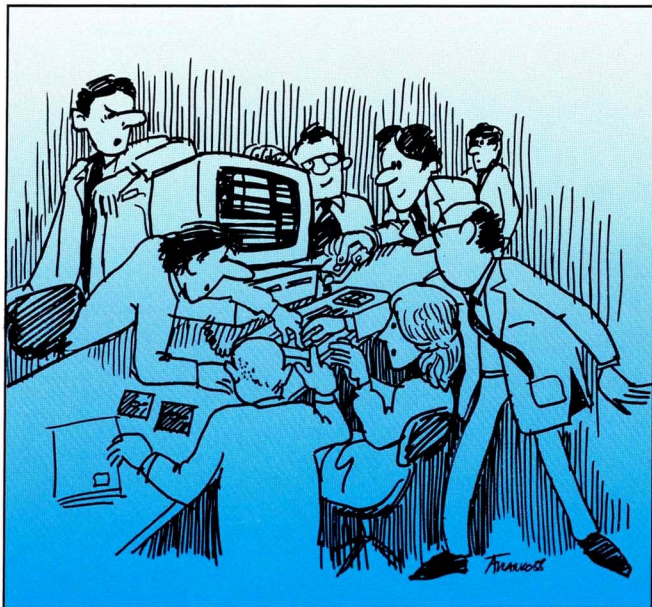
Atenção! Para já, só estão contratadas o inglês, o francês e o alemão. O castelhano e o português estão efectivamente previstos, mas para mais tarde.

Como já noticiámos, quatro dos dezasseis canais disponíveis no ASTRA, estão já contratados por Mr. Murdock, da Sky Television. O seu principal concorrente, Maxwell, por seu turno, reservou mais seis e é voz corrente que, até ao lançamento do satélite, em Novembro, existem 90% de probabilidades de todos os canais estarem ocupados.

A oferta é efectivamente óptima mas a disponibilidade de mais dezasseis canais de TV, mesmo em Inglês cuidado! ??? não é, ou pelo menos pode não vir a ser assim. É claro, que a Sky Television vai disponibilizar para recepção, no ASTRA quatro canais sem qualquer tipo de remuneração dos telespectadores mas, até quando? Por outro lado, os restantes canais vão transmitir em PAL ou DMAC? Poderão todos concluir que a publicidade não seja o único suporte económico-financeiro para o investimento e, daí, ser necessário cobrar de alguma forma, os serviços prestados, sempre numa base de descodificação do sinal emitido.

Dê por onde der, o que cada um pode entender é que as transmissões TV, via Satélite, desceram à Terra.

A sua acessibilidade vai ser uma realidade graças à AMSTRAD-FIDELITY.

OS PCW TAMBÉM PODEM SER TERMINAIS DO AMSTRAD MULTI-POSTO

A COMINFOR acabou de anunciar que as novas versões dos seus "KITS" multiposto K2000, K3000 e K4000 passam a poder suportar qualquer modelo de PCW como terminal das várias configurações de 2 a 4 postos de trabalho.

Como foi referido pela AM no seu número de Agosto, os PCW 8256/8512/9512 são capazes de incluir os terminais DEC VT100 e DEC VT52, atendendo a que o sistema operativo utilizado no AMSTRAD multi-posto — O PROLOGUE — é capaz de suportar terminais do tipo anteriormente citado, a COMINFOR acabou por acrescentar esta possibilidade aos emuladores que fazem parte do Kit multi-posto e, segundo nos informaram, sem qualquer aumento de preço.

Com esta nova possibilidade, qualquer pequena e média empresa pode dispôr de uma configuração com 4 postos de trabalho, impressora larga, 3 impressoras PCW e capacidade em disco de 20 MB, por pouco mais de 900 contos.

Como é do conhecimento geral, os PCW 8256/8512 têm capacidades de ecrã de 32 linhas por 90 colunas, superiores às habituais 25 linhas x 80 colunas dos ecrãs comuns. Funcionando no modo emulação, os PCW 8256/512 ficam com uma janela central onde se trata toda a informação à dimensão de 25 linhas x 80 colunas.

A COMINFOR garante ainda que todas as aplicações actualmente comercializadas para o sistema multi-posto funcionam com as mesmas teclas de função e de comandos que as utilizadas nos outros terminais.

AMSTRAD NÃO SÃO SÓ COMPUTADORES

A actual política comercial do líder de vendas de micro-computadores na Europa, aponta para uma diversificação das marcas que utiliza:

— AMSTRAD para os computadores profissionais
— SPECTRUM para os computadores domésticos
— FIDELITY para o ramo TV/Audio/Vídeo

Também nesta área, que, para já, marcará os seus produtos com AMSTRAD-FIDELITY, a AMSTRAD vai apresentar novidades no decurso do corrente mês.

Para além de um "camcorder" de fácil utilização e baixo preço, vai ficar disponível uma nova linha de Hi-Fi e vídeos com rácio performances/custo francamente fora daquilo a que estamos habituados no mercado.

Os tele-vídeos, conceito ainda novo no nosso país e que corresponde a um sistema integrado de televisão e vídeo serem outros produtos que vão ser comercializados, em breve, no nosso país.

A AMSTRAD ASSINA TROCA DE PATENTES COM A IBM

A AMSTRAD, PLC acabou de anunciar um acordo de troca de patentes com a International Business Machines Corporation (IBM). O acordo dá à AMSTRAD o direito de utilização das patentes IBM referentes a micro-computadores, para todo o mundo, incluindo as do IBM PC e as dos "IBM Personal System 2" — IBM PS/2.

O acordo, não excluindo para ambas as partes, dá também à IBM o direito de utilização de todas as patentes AMSTRAD.

Este contrato foi designado, por ambas as partes, como de longa duração, tem efeito a partir de 1 de Julho do corrente ano e supõe-se que o término seja em Junho de 1993, ou seja, terá uma duração de 5 anos.

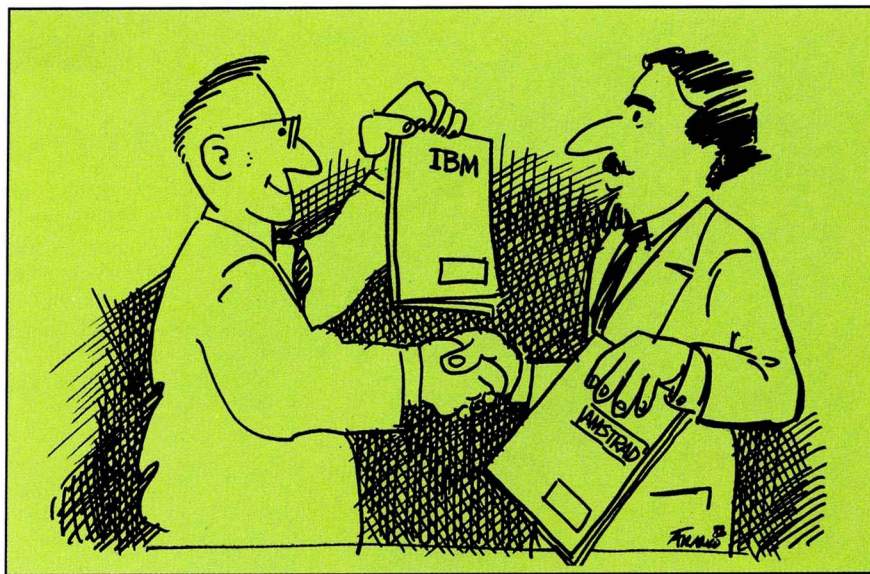
No decorrer da conferência de imprensa dada sobre este acontecimento, Alan Sugar disse estar a AMSTRAD muito agradada com o acordo celebrado com a IBM, especialmente por ter sido sempre política da companhia respeitar os direitos de autor dos outros fabricantes e esperar continuar, a AMSTRAD, a seguir tal política.

O Presidente da AMSTRAD disse ainda que, se existissem dúvidas ou incertezas quanto à legalidade dos computadores profissionais que tem vindo a comerciallizar nos últimos anos, essas ficariam completamente dissipadas após a assinatura deste acordo.

A AM convidou a COMINFOR a pronunciar-se mais promenorizadamente sobre as consequências deste acontecimento, e obteve a quase garantia do lançamento de compatíveis IBM PS/2 para o final do corrente ano.

O ACORDO AMSTRAD/IBM VISTO POR DENTRO

A indústria de computadores ficou altamente surpreendida com o acordo ainda há pouco assinado entre a AMSTRAD e a IBM, especialmente por se tratar de uma troca de patentes, sem encargos de parte a parte — é a primeira que o afirma — em vez duma concessão de direitos de utilização, devidamente remunerada.

Efectivamente parece difícil de aceitar um contrato deste tipo entre duas empresas que se podem considerar de tipos completamente opostos. Por um lado, a AMSTRAD que comercializa computadores a metade dos preços habituais e, entre eles, super-cópias dos IBM-PC. Por outro lado, o gigante da indústria dos computadores, que tem relutância em admitir os baixos índices performance/preço no campo da micro-informática e que baseia toda a sua estratégia de venda no seu indubitável prestígio e reputação.

Pelo menos na Europa, não parece que a AMSTRAD tenha contribuído para resolver o problema do abaixamento de vendas dos micros IBM.

Mas, para aqueles que pensam que este acordo foi outro erro da IBM, deixemos completamente claro que é capaz de não ter sido tanto como isso.

A "press release" da AMSTRAD diz que "o acordo celebrado garante à AMSTRAD uma utilização, não excessiva, mas para todo o mundo das patentes da IBM, incluindo as dos PC, PS/2 e arquitectura micro channel". Tal como foi escrito, tudo leva a crer que a AMSTRAD pode fazer, a partir de agora "cópias" dos PS/2 e comercializá-las mais baratas — sem qualquer espécie de dúvida.

A posição da IBM é, no entanto, bastante antagónica. Um responsável pelas relações públicas da companhia diz que os direitos concedidos para utilização da MCA (Micro Channel Arquitecture) não permitem à AMSTRAD fazer cópias integrais dos PS/2. Acrescenta ainda que, os mesmos direitos já teriam sido concedidos a outros construtores, nomeadamente a COMPAQ, e que tudo seria bom para o desenvolvimento que cada um deveria fazer.

Sugar, ao afirmar que "o contrato não nos dá direito a tudo copiar, mas cá temos os técnicos para conceber o que faltar, tal como sempre o fizémos", não está nuito longe da posição da IBM.

Por muito mais que se diga, achamos que as peças somente foram posicionadas no tabuleiro de um grande xadrez. É óbvio que a AMSTRAD não disse, nem escreveu, o que pretende obter deste acordo. Por parte da IBM, supõe-se haver vontade em recuperar, à custa deste contrato, algumas licenças perdidas sobre os computadores que a AMSTRAD tem vindo a comercializar massivamente.

Ou será que o "Big Blue" já entendeu que já existe muita gente neste mercado de micro-informática e que "tudo o que vier à rede é peixe", seja em termos do licenciamento de patentes, seja em termos de utilização de patentes de outros?

Só o tempo, que não vai ser longo, poderá esclarecer muitas destas questões.

AMSTRAD=SUCESSO

A elevada taxa de crescimento da AMSTRAD PLC nos últimos anos chamou a atenção da AM que, assim, se dispõe a dar uma panorâmica dessa evolução aos leitores.

Como é do conhecimento geral, 1987 foi o primeiro ano completo de vendas dos PC AMSTRAD. Foram cerca de 360 000 unidades que contribuiram para um volume de negócios global de 130 milhões de contos, com um lucro bruto de 35 milhões.

Com uma facturação no primeiro semestre de 1987 semelhante à do primeiro semestre de 1986, o ano terminou com um volume de negócios superior em 20% ao do ano anterior, como se pode ver através do gráfico na fig.1.

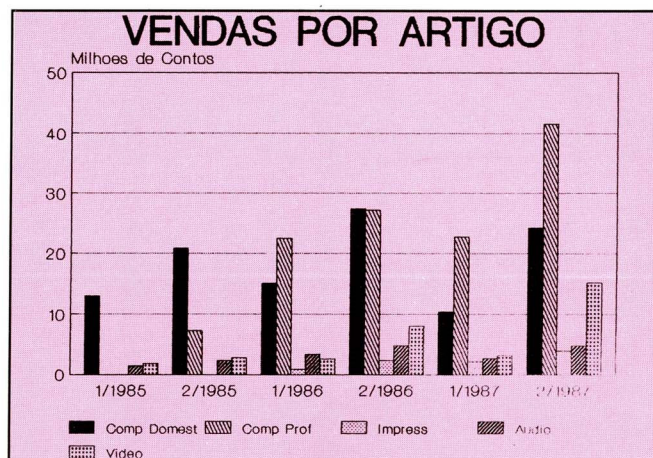
Para além do que foi referido, os resultados têm-se mantido desde o início em ascensão, de uma forma proporcional à facturação global da empresa.

Como tem conseguido a AMSTRAD a taxa de crescimento a que nos tem habituado? A resposta é simples: através de uma busca sistemática de novos artigos de grande divulgação, sem deixar cair ou substituir a gama disponível.

É disso que nos dá conta o gráfico apresentado na fig.2, onde se constata o contínuo crescimento da empresa na maioria dos mercados que aborda: Antes de 1985, exclusivamente ligada à linha de audio; em 1985 lança as bases no domínio dos computadores domésticos, dos vídeos e dos computadores profissionais; e em 1987, o mercado das impressoras passa já a ser representativo.

Aliada à estratégia do aumento da gama de produtos comercializados, a AMSTRAD tem também desenvolvido os mercados onde coloca esses produtos.

E, desde 1983 (ano em que a exportação da marca era praticamente nula e toda a sua actividade era dirigida para o mercado inglês) até 1987, (ano em que a exportação ultrapassa os 60%) fica um longo caminho que foi muito rapidamente percorrido.

E a que se deve tal sucesso? Por muito eficazes que sejam as organizações montadas, não tenhamos dúvidas que os líderes são o ponto fulcral das empresas. A AMSTRAD não é excepção e Sugar é a chave do êxito. Sir Clive Sinclair, o homem que muitos consideram ser o pai dos computadores domésticos, um dia descreveu Sugar: "Absolutamente brilhante — um génio a trabalhar no seu próprio campo". E continuou: "a coisa que faz sempre correctamente é entrar nos mercados na altura oportuna e também sabe sair deles oportunamente".

ALAN SUGAR — UMA ANÁLISE DA SUA PERSONALIDADE

O líder da AMSTRAD, é um homem cuja brusquidão no contacto com os seus colaboradores é famosa. "Ninguém se safa" — Dentro da AMSTRAD esta nova expressão está gravada em todos para descrever a situação das pessoas tremerem debaixo do olhar de Alan Sugar.

Esta sem-cerimónia é interpretada de várias formas: rudeza, excentricidade, dinamismo. Qualquer que seja o caso aplicável a Sugar; não escapa a suspeição de que ele cultiva isso deliberadamente. É uma forma de evitar perdas de tempo.

Uma das pedras de toque da personalidade de Sugar e base do seu sucesso é a sua perfeição na arte de administrar o tempo. Praticando este tipo de austeridade, Sugar consegue esta rara dupla faceta: dirige um negócio altamente bem sucedido, motivando ou aterrorizando o seu staff (dependendo do ponto de vista de cada um) sem comprometer a sua vida privada. Trabalha relativamente poucas horas, já que a administração do tempo permite-lhe dedicar grande parte dele à sua mulher e três filhos.

Sugar não é o ignorante "barrow-boy" que o mito de Londres-Este fez crer. Ele deixou a escola com suficientes níveis A para poder ingressar na faculdade, mas preferiu dedicar-se àquilo para que provou ter um particular talento — a arte de fazer dinheiro.

O sucesso da carreira de Sugar nos negócios é visto aliado à coragem das suas decisões nos momentos cruciais. O primeiro talvez tenha surgido quando convenceu a rede de distribuição Comet a aceitar o equipamento Hi-Fi AMSTRAD. Mais tarde, quando os rivais japoneses começaram a concorrer em preço com o Hi-Fi AMSTRAD, lançado produtos baratos e frágeis, o primeiro computador AMSTRAD, o 464, veio a ter uma carreira de sucesso. O lançamento do processador de textos AMSTRAD PCW 8256, não foi apenas uma peça inspirada numa leitura rápida; ele deu a Sugar o meio de que necessitava para mais tarde competir com a toda poderosa IBM no negócio dos computadores pessoais.

Os seus instintos continuaram a ser vitais para a AMSTRAD. A forma como a companhia se desenvolveu e se apresenta no mercado mostra o seu rosto. AMSTRAD é Alan Sugar. A identificação não podia ser maior. "Ele começa cada corrida de lebre naquele lugar", diz um recente associado.

Um estudo de uma revista sobre os novos computadores AMSTRAD concluiu — "Se a AMSTRAD puder convencer o mercado que o PPC 640 é suficientemente barato então esta empresa terá outra venda espectacular nas suas mãos". E aqui outro mito de Sugar vem para ficar.

Ele gosta de apresentar uma imagem de uma direcção cuidada nos caminhos da City e, assim, não existem muitos projectos lançados pela AMSTRAD sem que as consequências financeiras não tenham sido antecipadamente cuidadas e medidas. Mas, quando acontecer que esses projectos sejam criticados, Sugar é ainda capaz de absorver informação de forma suficientemente rápida para embaraçar supostos especialistas, subordinados ou associados.

A proeminência de Sugar tornou-se notada nos anos Teatcher e, para alguns, ele é o emblema dos conservadores modernos. De espírito empreendedor e agressivo, Alan Sugar possui grande habilidade para dominar as forças que lhe são adversas.

E se ele não é propriamente um cavalheiro, tal só acontece porque os métodos AMSTRAD não são totalmente britânicos. Existem aspectos na política de Alan Sugar que são alheios ao contexto do "United Kingdom":

Primeiro: Ele não é enfaticamente um inventor. Os produtos AMSTRAD provêm das visitas de Sugar ao Extremo Oriente onde ideias e tecnologias estão disponíveis para serem copiadas.

Segundo: Os seus produtos são feitos no Extremo Oriente, onde a mão de obra é muito barata.

Terceiro: As operações de Marketing Internacional são altamente eficazes. Sugar tem muita destreza para os mercados de invenções.

O primeiro computador AMSTRAD CCP 464 deveria ser originalmente qualquer coisa que os favoritos já o fossem (dos sinclair ao Acorn). Esperavam-se micros que lhes disputasse o mercado, mas, em vez disso, ele criou um novo mercado entre os antigos.

A promoção dos computadores pessoais AMSTRAD no mercado mais difícil, o Americano, foi um típico empreendimento ousado. O primeiro anúncio parodiou toda a família americana, sugerindo o que 'Mom', 'Pop' e 'Júnior' (Mãe, Pai e Filho) poderiam fazer com o AMSTRAD PC. Todo o mercado pode ter sido "ofendido", mas existem agora sinais de que a AMSTRAD teve sucesso naquilo que foi o cemitério das grandes companhias inglesas de computadores.

A maioria dos fornecedores esforçam-se por conseguir conciliar duas exigências: a tecnologia avançada que as pessoas podem querer no futuro, e a necessária tecnologia que renda hoje um proveito efectivo para eles. Sugar experimentou e testou tecnologias, fabricando produtos que o público comprador descobria para a sua surpresa, que era, aliás, aquilo que sempre quis.

O futuro sucesso da AMSTRAD depende praticamente de Alan Sugar. O seu grande progresso mostrou, de certo modo, que o seu ponto de vista era correcto e as ópticas alternativas foram gradualmente postas de lado.

Não existem obstáculos para Sugar na AMSTRAD, apenas o desenvolvimento de uma cultura colectiva modelada na sua personalidade.

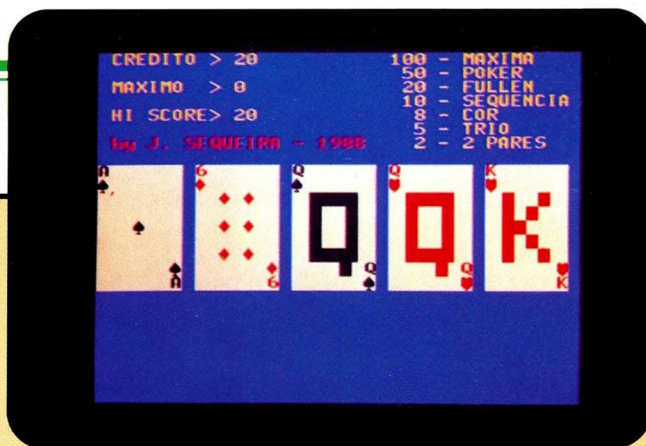
A NOVA REDE DA AMSTRAD

A AMSTRAD acabou de assinar com a CORVUS Systems Limited um acordo para desenvolvimento de uma versão da CROVUS Omninet a ser comercializada exclusivamente pela AMSTRAD com a designação AMS-NOS, pelo menos, na Europa. Poderá acontecer que a rede venha a ter outro nome no mercado dos Estados Unidos.

Faz também parte do acordo o desenvolvimento de modificações no PC-NOS, propriedade da CORVUS, para adaptar à nova rede.

O negócio pode vir a trazer à CORVUS um acréscimo na sua facturação de cerca de 3 ou 4 milhões de dólares.

Espera-se, entretanto, que a versão AMSTRAD da CORVUS Omninet esteja disponível a partir do terceiro trimestre deste ano.

O preço previsto para um KIT de 3 postos, em Inglaterra, é de cerca de 100 contos.

OS HOBBIES DO ENG.º SEQUEIRA

O POKER, jogo que a AM tem vindo a comercializar no Clube dos Leitores, é um verdadeiro exclusivo nosso.

Conseguimo-lo pelo facto de, numa das "secas" a que já estamos habituados a "levar" quando pretendemos contactar os responsáveis pela COMINFOR, a nossa atenção ter sido despertada pelo referido jogo que se encontrava em demonstração.

A semelhança, quer em imagem, quer em som, com o das máquinas que há bem pouco tempo intentaram proliferar em cafés, "pubs" e "boites" e que hoje recolheram aos casinos, era francamente notável.

Interrogámos alguém da Cominfor sobre a origem e preço do jogo e responderam-nos que não era para vender. Tratava-se de um trabalho feito em férias pelo Eng.º José Sequeira, Director Geral da Cominfor, cujo "hobby" é fazer programas.

E esta, hem?!!!

AMSTRAD PS/2

E Tudo começa com: A FÚRIA DE SUGAR

PARA quem conhece a política da AMSTRAD no que respeita ao sigílio que mantém quando ao lançamento de novos produtos na área da informática, é fácil imaginar a irritação de Alan Sugar ao tomar conhecimento das afirmações feitas por José Luís Dominguez, presidente da AMSTRAD Espanha, em entrevista concedida ao Tribuna Informática.

José Luís Dominguez teria afirmado que, em Setembro, no PC SHOW em Londres, a AMSTRAD iria lançar uma gama de clones PS/2, baseados nos INTEL 80286 e 80386, com arquitectura MCA (Micro Channel Arquitecture) e o standard gráfico VGA.

O NOVO PRODUTO

Por conhecer um pouco a personalidade do líder da AMSTRAD Espanha, que põe sempre um certo exagero nas suas afirmações, a AM não considerou possível que a AMSTRAD, viesse a lançar para já um PS/2 compatível.

Em primeiro lugar, porque é política da companhia escolher um mercado de grande divulgação com tecnologias já altamente experimentadas e banalizadas, consequentemente mais baratas. O lançamentoo de um compatível PS/2, ao nível alto da gama seria uma inversão desta política, o que não nos parece estar para acontecer.

Em segundo lugar e por muito que o tema já tenha sido elaborado não nos consta que a IBM ou a INTEL já tenham licenciado outros construtores a fabricar o INTEL 80386 que suporta os PS do alto da gama. Assim, a AMSTRAD teria que comprar os referidos processadores a preço INTEL e a característica de baixo preço dos seus produtos ficaria assim abandonada.

Em terceiro lugar, a AM, ainda há bem pouco tempo só conhecia o licenciamento da arquitectura MCA, no que se refere a política comercial, isto é, aqueles que a fundamentam mais numa base da tradição e prestígio (Compaq, Olivetti e Dell) do que no preço (AMSTRAD). A "guerra" que a AMSTRAD lançou há conquista de, pelo menos, o mercado europeu, com a vitória que não se lhe pode consertar, faz tornar ainda mais imprevisível o licenciamento de patentes IBM a este "agressor".

Em quarto lugar, a AMSTRAD tem por hábito apoiar os seus produtos com software adquirido aos seus conceptores que comercializa a preços "AMSTRAD". Ora ainda não existem no mercado "packages" a correrem sob OS/2 que possibilitem a continuidade dessa política.

Por tudo o que foi dito, o mais que a AMSTRAD poderia fazer seria lançar um novo modelo do PC 1640 com diskettes de 3"1/2 que seria compatível com o IBM PS/2-30. Isto constituiria a primeira grande desilusão que a AMSTRAD daria aos seus habituais consumidores dado o PS/2-30 com o processador 8086 nada ter de diferente de qualquer PC com o referido processador e diskettes 3"1/2.

A AMSTRAD MAGAZINE está pois convicta que para o fim do ano a AMSTRAD vai lançar um novo computador para complemento da sua gama e que este será um compatível AT com elevada velocidade e diskettes de 3"1/2. Este modelo seria similar ao PS/2 da gama média distinguindo-se destes por utilizar tecnologia "bus AT" em vez da MCA.

OUTRA FRENTE DE ATAQUE

A AMSTRAD MAGAZINE está ainda convencida que o final do ano será também a data escolhida pela AMSTRAD para lançamento de um outro computador na área dos micro-domésticos.

Ao contrário do que já tem sido dito por muitos, não será O SPECTRUM+4 compatível Amiga que vai ver a luz do dia no final deste ano, nem sequer nos tempos mais próximos.

Pensamos que a AMSTRAD vai lançar uma máquina de jogos compatível PC. O sistema seria um compacto de 16 bits com uma ou duas diskettes de 3"1/2 integradas no teclado. Toda a sua arquitectura interna seria baseada no PC 1512 e admite-se que o seu preço venha a rondar os 80 contos, sem monitor. O modelo permitiria a utilização, tal como nos SPECTRUM+2 e +3, de uma vulgar TV para funcionar como ecrã, o que não quer dizer que a AMSTRAD não venha também a comercializar monitores próprios, para estes equipamentos. E porque não "SPECTRUM+4".

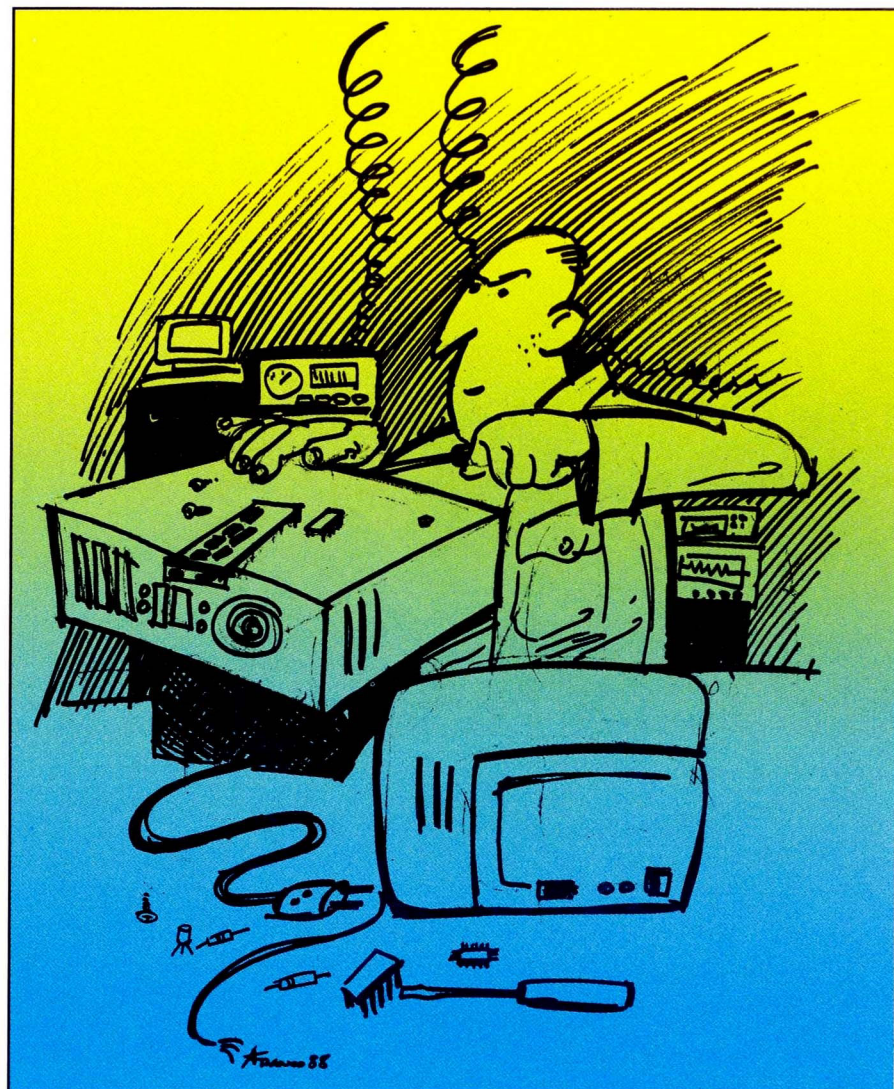
Como já foi amplamente divulgado a estratégia comercial da AMSTRAD, PLC aponta para a divulgação dos seus produtos, utilizando marcas distintas: SPECTRUM para os computadores domésticos, FIDELITY para o audio e vídeo, que adquiriu a seu tempo aos seus detentores. O nome AMSTRAD contínua e continuará a marcar os produtos que mais lançaram a companhia: os computadores profissionais.

No entanto e pelo menos nesta fase, o doméstico de 16 bits, não correrá os programas até agora desenvolvidos para SPECTRUM e, daí, a não utilização desta marca para evitar que fossem criadas confusões no mercado.

O ACORDO AMSTRAD/IBM

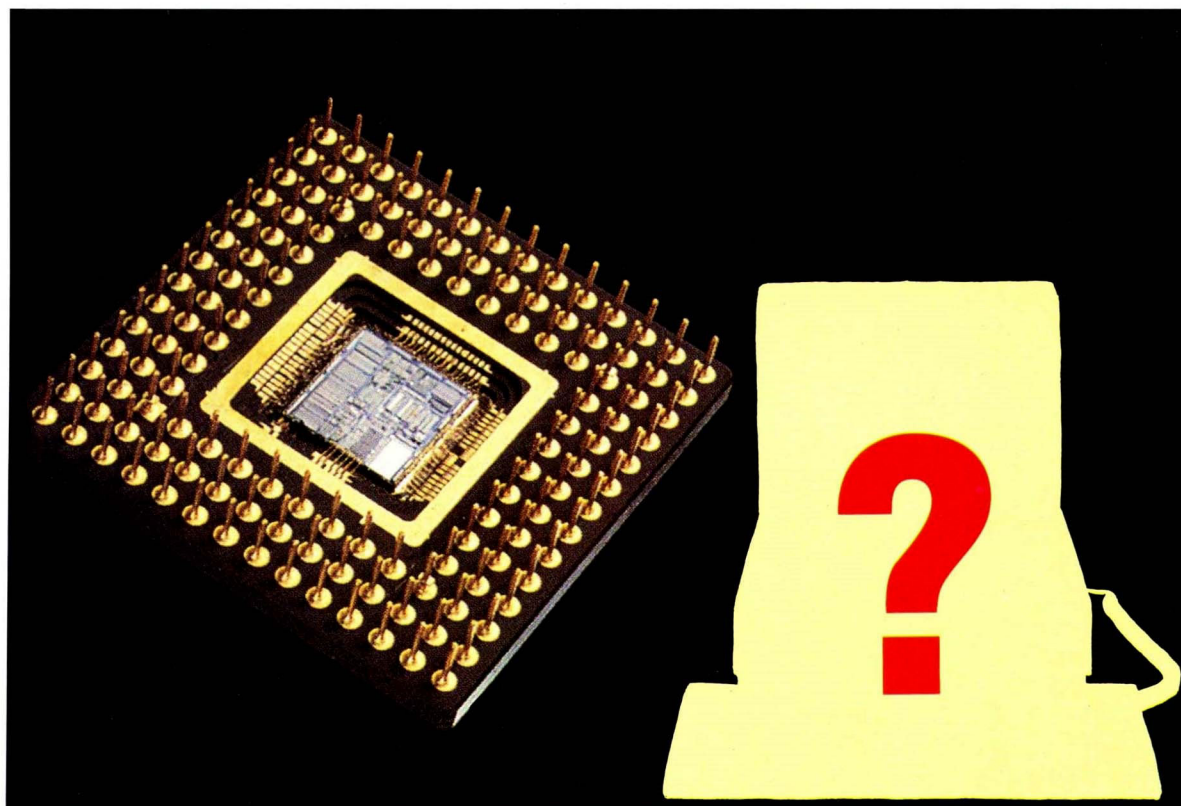
Tudo o que anteriormente foi dito tem lógica. É a opinião da AM, que coincide com a de muitos analistas da evolução de equipamentos de informática.

Mas ainda há bem pouco e tal como noticiamos com pormenor, noutra secção da AM, a AMSTRAD assinou com a IBM uma troca de patentes que lhe permite a fabricação de compatíveis PS/2 com tecnologia MCA.

Por muito que a AMSTRAD tenha vindo a divulgar tal acto e por muito pouca importância que a IBM lhe tenha dado, o que é um facto pouco previsível é que a AMSTRAD o tenha feito somente para ter um pico muito afiado na cotação das suas acções na bolsa, isto é, a AMSTRAD vai tirar proveito deste acordo.

O PORQUÊ DO AMSTRAD PS/2

A opinião avançada anteriormente tem argumentos capazes de o suportarem. Em primeiro lugar, o acordo celebrado entre a AMSTRAD e a IBM deve ter custado à primeira alguns milhares de libras e não nos parece que tal tenha sido feito numa óptica de ganhar prestígio. Seria um forte investimento com vida curta, em breve, revendedores, accionistas e utilizadores da AMSTRAD sentir-se-iam defraudados pela concretização da imagem que o referido acordo lhes teria suscitado.

Em segundo lugar, a AMSTRAD tem que manter o seu crescimento meteórico. Os interesses envolvidos já não permitiriam a Alan Sugar baixar as taxas de crescimento da empresa que dirige e por outro lado, a empresa ainda não solicitou a posição a que se guindou nos últimos anos, para encarar o futuro numa base do "vamos andando, porque todos estão connosco". Este imprescindível crescimento não se pode conseguir por um aumento das vendas dos produtos existentes. Há que alargar a gama e os PS/2 poderão ter bons apoios neste caminho.

Em terceiro lugar e, como já tem sido referido na AM, a AMSTRAD ainda não conseguiu o lugar que deseja a nível das grandes companhias. O PC 1640 seria o equipamento destinado a esta área de mercado, mas a sua penetração a nível das grandes empresas, banca e seguros parecem ainda não satisfazer a AMSTRAD. Assim um equipamento com as novas características apresentadas no mercado (VGA, MCA) seria um salto no prestígio da AMSTRAD e a grande oportunidade para se integrar nestes apetecíveis mercados.

A AM, por estas razões acha que a AMSTRAD vai lançar também compatíveis PS/2 num curto espaço de tempo e isto, sem discurar a evolução da linha PC.

O QUE SERÁ O PS/2 DA AMSTRAD

A aquisição de direitos sobre a arquitectura MCA não de direito, só por ela, a fazerem-se cópias integrais dos modelos PS/2. Tal foi afirmado por um alto dirigente da IBM e é mais ou menos óbvio.

Tais direitos permitem, tal como a Compaq o fez, fabricarem-se modelos "like PS/2" normalmente com maiores perfomances e a menores preços, pelo menos no caso da AMSTRAD.

Diskettes de 3"1/2, capacidade de correr o OS/2 e arquitectura MCA são indubitavelmente características destes novos modelos da AMSTRAD.

Ficam, no entanto, algumas dúvidas, quanto a processador e aos ecrãs disponíveis.

Enquanto defendemos a possibilidade do lançamento de um AMSTRAD baseado no INTEL 80386, compatível PC, achamos que na gama PS, a AMSTRAD não "atacará" os compatíveis PS/80 da IBM, isto é, ficar-se-á pelos equipamentos baseados no 80286. Fundamentalmente, pelo preço. Existem dois factores que fariam o "AMSTRAD PS/80" ser caro. O 80386 a ser comprado à INTEL a adicionar aos direitos de arquitectura MCA. O preço saíria bastante dos "standards AMSTRAD" e os consumidores interpreta-lo-iam como se a marca "já estivesse como os outros".

Por outro lado, o OS/2 que tem constituído handicap positivo para os PS/2 foi concebido especialmente para ser suportado pelo 80286. Não haveria pois vantagem no lançamento de um compatível PS2/80.

Quanto aos seus ecrãs não duvidamos terem sido o "calcanhar de Aquiles" da AMSTRAD. Os utilizadores exigem cada vez mais maior resolução e possibilidades de ecrã — é aquilo que veêm. A AMSTRAD só atenuou a sua má imagem neste domínio com o lançamento dos PC 1640 HERCULES (monocromáticos) e EGA, e não foi por certo a tecnologia LCD dos PPC que a ajudou neste domínio. O dinamismo irrefutável da AMSTRAD aponta para que haja necessidade de dar a volta. É opinião da AM que para o compatível PS/2 estarão disponíveis vários tipos de ecrá incluindo os VGA, que possuem a maior resolução entre os novos standards do mercado.

CONCLUSÃO

A AMSTRAD, desde que se instalou no mercado dos computadores e pelo dinamismo que tem imprimido à sua divulgação, gera sempre curiosidade de especulações sobre aquilo que vai ser a nova "bomba AMSTRAD".

A AMSTRAD MAGAZINE não acerta sempre, mas tem sempre como preocupação deixar os seus leitores bem informados através de análises correctas do mercado.

Neste estilo, a AM deixa claras as suas convicções para o Outono deste ano no que diz respeito à evolução dos produtos AMSTRAD no âmbito da Informática:

Vai ser lançado um computador doméstico de 16 bits, compatível com IBM PC, a diskettes de 3"1/2 capaz de se conectado aos tradicionais aparelhos de TV.

Os actuais PC 1512/1640 não vão acabar e verão a sua gama alargada com a introdução de um equipamento suportado pelo INTEL 80286, de 16 bits, com elevada velocidade do relógio interno.

É mesmo possível que venha a ser lançado um micro na base 80386, sem arquitectura MCA.

Os compatíveis PS/2 do médio da gama (com processador INTEL 80286) podem também surgir.

Se atendermos aos já divulgados PCW 8256/8512/9512, aos PC 1512/1640, aos portáteis PPC e aos produtos que a AM prevê em vias de aparecimento no mercado, a AMSTRAD fica efectivamente com a maior gama de micros que, alguma vez, qualquer fabricante tornou disponível.

LINGUAGENS DE PR

Desde o princípio da era da informática, cientistas e programadores têm tentado simplificar a relação entre o homem e o computador, criando diferentes linguagens de programação. Mas, quais são as melhores? Porque é que existem tantas?

OGRAMAÇÃO

IMAGINEMOS por instantes que necessitamos explicar a alguém o modo de marcar um número de telefone e atender uma chamada. Ainda que possamos descrever o processo passo a passo, clarificando até os detalhes mais insignificantes, o mais provavel é que não façam falta mais que umas quantas instruções de caracter genérico. Contudo, se o nosso hipotético interlocutor não fosse uma pessoa mas sim um computador, haveria que indicar-lhe todos os pormenores da operação. O mesmo ocorreria com qualquer outra actividade, pois, tecnicamente, a unica forma de conseguir que um computador faça algo é processar isso convenientemente, ajustando milhares de comutadores microscópicos de duas posições: aceso e apagado.

A programação do computador ao mais baixo nível,é feita através de códigos binários (zeros e uns) que controlam os citados comutadores, permitindo executar tarefas muito simples, como por exemplo, somar os números ou armazená-los na memória. Quando várias centenas ou milhares destes códigos elementares se combinam para formar um programa , o computador parece executar tarefas de grande complexidade, ainda que, na realidade, esta seja uma falsa aparência.

Programar o computador à base de códigos binários, quer dizer, em linguagem máquina, é um trabalho desalentador. Para fazer esse trabalho de uma forma mais fácil e mais rapida , durante as últimas quatro décadas tem surgido uma infinidade de linguagens de programação, empregando quase todas palavras inglesas como elo de ligação entre o programador e a máquina.

Assim, uma das instruções típicas da linguagem Basic (Beginner's All Purpose Symbolic Instrution Code) é PRINT, cuja função é exactamente o que o seu nome sugere. À execução da ordem PRINT "Olá" , o computador imprimirá no ecrã o texto entre aspas. Para obter o mesmo resultado em linguagem máquina, provavelmente iriam ser necessários mais de meia dúzia de comandos.

As linguagens como o BASIC denominam-se de alto nível. A diferença entre elas e o código máquina é semelhante a dizer a alguém: "Por favor, atende o telefone", ou explicar-se passo a passo, sem esquecer sequer os detalhes mais irrelevantes, a forma de atender uma chamada. As linguagens de baixo nível, como o código máquina ou o assembler, têm a vantagem da sua alta velocidade de execução e de permitirem um controlo directo do hardware, mas são de difícil aprendizagem e utilização. Contrariamente, as de alto nível são sempre mais lentas, mas são quase independentes do hardware e de fácil aprendizagem. As operações em que o assembler requer dezenas de instruções são normalmente obtidas através das linguagens de alto nível e com poucos comandos.

PRÉ-HISTÓRIA DAS LINGUAGENS

Mas porque é que existem tantas linguagens de alto nível diferentes? Em primeiro lugar, porque a diversidade de aplicações do computador assim o exige; uma linguagem válida para escrever um programa de contabilidade possívelmente não seria a mais adequada para um jogo de aventuras. Em segundo lugar, como cada pessoa tem o seu estilo próprio de programação, quantas mais linguagens houver, mais possibilidades teremos de encontrar uma que se adapte ao nosso caso.

O constante desenvolvimento das linguagens de programação de alto nível, distanciou os programadores dos computadores, de forma que, cada vez mais, eles ignoram o que ocorre no interior da sua máquina. É como conduzir um automóvel sem ter conhecimentos de mecânica: ainda que possa nunca fazer falta, se acontecesse uma avaria comprender-se-ia a sua importância.

Actualmente é possível executar um programa em qualquer computador sem saber absolutamente nada de programação. Basta inserir uma disquete ou uma cassete, ligar a máquina e, quanto muito, digitar um fácil comando. Mas nem sempre foi assim.

Retrocedamos quatro décadas para nos situarmos nos tempos do primeiro computador electrónico digital, o ENIAC (Electronic Numeric Integrator and Calculator), um colosso de trinta toneladas de peso construído durante a Segunda Guerra Mundial na Universidade da Pensilvânia, com mais de dezoito mil válvulas electrónicas de vácuo e dezenas de milhar de resistências e condensadores. A programação do ENIAC consistia , por surpreendente que pareça, em alterar a um conjunto de cabos, de cada vez que se pretendia executar um novo programa, os engenheiros encarregavam-se de mudar adequadamente as ligações da máquina. O computador carecia de uma memória que lhe permitisse armazenar programas e as unidades de disco eram inimagináveis. Como consequência, só um grupo restrito de cientistas e de engenheiros poderiam realmente "comunicar" com o computador.

A construção e manutenção destes arcaicos computadores era extremamente dispendiosa. Para cúmulo, a sua rudimentar forma de programação obrigava a perder um valioso tempo, de cada vez que se mudava de programa.

Felizmente, que se levou rapidamente à pratica uma das ideias de John Von Neuman: o armazenamento dos programas. Um computador com uma memória capaz de armazenar os programas enquanto se executavam, seria muito mais eficaz que o primitivo ENIAC. Para mudar de programa bastaria simplesmente substituí-lo por outro em memória.

Em 1948, construiu-se no Reino Unido, o Mark I, o primeiro computador capaz de armazenar programas e programável mediante a manipulação de um conjunto de comutadores situados na zona frontal. Foi um grande avanço, mas todavia insuficiente. Para mais, os programas eram introduzidos ao contrário, começando pelo final.

O passo seguinte foi encontrar um sistema para conservar permanentemente os programas e os dados, evitando ter de "teclá-los" de cada vez que se fossem utilizar. Nessa altura, seguiu-se uma ideia elaborada pelo matemático do século passado Charles Babbage, que por sua vez se inspirou num mecanismo utilizado em França para o controlo de teares: os cartões perfurados.

Os cartões perfurados possibilitavam a escrita e o armazenamento de programas independentemente do computador. Os programadores escreviam os seus programas com a ajuda de máquinas perfuradoras e esperavam pela sua vez para introduzir os cartões no computador, o qual, antes de imprimir os resultados, ficava disponível para receber um novo lote de cartões. Este sistema era chamado o processamento por lotes.

Pela primeira vez os programadores estavam fisicamente separados dos computadores. Existiam especialistas em software, que escreviam os programas em lotes de cartões, e especialistas em hardware, encarregados da manutenção do equipamento e de colocar os programas em execução.

Apareciam nessa altura os programas assembladores, que traduziam mnemónicas como LD HL, DE (carregamento do registo HL com o conteúdo de DE) pelo código binário de zeros e uns compreensível pelo computador. Cada código de operação do assemblador correspondia directamente a uma instrução de linguagem máquina.

Rapidamente os programadores começaram a reunir pequenos programas escritos em Assembler. Deste modo, se alguém necessitasse de uma rotina para calcular raízes quadradas, podia pedi-la emprestada a outro programador que a tinha escrito anteriormente. As bibliotecas de funções foram o primeiro passo no sentido das linguagens de alto nível.

LINGUAGENS DE ALTO NÍVEL

Uma das primeiras linguagens de alto nível foi o FORTRAN (FORmula TRANslator), desenvolvido em 1954. Antes do seu aparecimento, cientístas e engenheiros não familiarizados com os computadores tinham que descrever o problema que desejavam analisar a um programador, que se encarregava de escrever um programa para solucioná-lo. O FORTRAN foi criado com a intenção de facilitar aos cientistas e aos engenheiros a escrita dos seus próprios programas, sendo ainda hoje muito utilizado em universidades e centros de investigação.

Do mesmo modo que o FORTRAN se desenvolveu para aplicações científicas, o COBOL (COmmon Business Oriented Language) surgiu em finais da década de cinquenta com o propósito de facilitar a escrita de aplicações comerciais, como contabilidade, stocks, etc.. Ainda que seja frequentemente criticada, é no entanto, a linguagem mais utilizada nos grandes computadores destinados a tarefas de gestão.

Em 1960 apareceria o ALGOL e o LISP, linguagens de características muito diferentes. A primeira é muito semelhante ao FORTRAN e caracterizou-se como sendo uma linguagem de aplicação geral. A segunda está orientada para o processamento de conjuntos de elementos simples, dados de uma forma específica. Pela sua capacidade para o manuseamento de símbolos, LISP (LISt Processor) utiliza-se em investigações de inteligência artificial, manipulação algébrica, análise linguística e aplicações similares. A diferença está que enquanto o FORTRAN e o COBOL se consideram obsoletas, o LISP se mantêm ainda pujante.

Em 1964, dois professores da Universidade de Dartmouth, John Kemeny e Thomas Kurtz, formaram uma dupla de oonde surgiram excelentes ideias. A primeira foi a sugestão de que em vez de processar os programas por lotes, os computadores deveriam ter vários terminais ligados, de forma a poderem repartir o seu tempo entre diferentes utilizadores.

Actualmente, é possível com qualquer computador, executar programas sem saber absolutamente nada de programação.

Um sistema de terminais de tempo repartido permitiria que várias pessoas utilizassem simultaneamente um computador (sistema multiutilizador e multitarefa). A segunda ideia tinha a ver com uma nova linguagem de programação de carácter geral, mais fácil de aprender que o FORTRAN ou o COBOL e muito mais fléxivel: o BASIC. Esta linguagem caracterizou-se desde o princípio pela sua interactividade com o utilizador e pela sua sintaxe pouco rígida. Pelo seu reduzido tamanho e facilidade de implementação, é normal entregar-se com a maioria dos microcomputadores. O BASIC e os sistemas de terminais em tempo repartido ultrapassaram os sistemas de processamento por lotes, simplificando o trabalho dos programadores.

Depois do BASIC, nasceram muitas outras linguagens. Os computadores empregavam-se cada vez mais para aplicações específicas e passou a existir uma grande procura de linguagens de programação mais especializadas.

Nos finais da década de sessenta, o tema da actualidade no mundo informático era a programação estruturada. Trata-se de um método destinado a tornar os programas mais legiveis e a facilitar correções e modificações. A primeira linguagem desenhada para ajudar os programadores a programar de uma forma estruturada foi o Pascal, criado em 1971 por Niklaus Wirth. O Pascal conta com grande implantação em meios académicos, sendo considerada uma linguagem ideal para se adquirir um bom estilo de programação.

Entretanto, a informática tinha começado em alguns países a ensinar-se nas escolas, inclusivé nas primárias, e havia necessidade de dispor de uma linguagem de programação adequada aos mais jovens. Nos finais dos anos sessenta, Seymour Papert do MIT (Massachussetts Institute of Technology) concebeu e desenvolveu o LOGO, que, em certa medida é um dialecto do LISP. Muitos dos comandos LOGO servem para instruções a uma tartaruga ou cursor, cujos movimentos permitem traçar gráficos. O LOGO é uma linguagem interactiva e interpretada.

Actualmente, uma das linguagens a ganhar "terreno" é a linguagem C, que data do início dos anos setenta. Desenvolvida inicialmente para o sistema operativo UNIX por Dennis Ritchie, trata-se de uma linguagem de nível intermédio, versátil, flexível, veloz e transportável. O estilo dos programas escritos em C é semelhante ao do Pascal, ainda que, ao não existirem apenas restrições, o programador deve ter mais cuidado em obter um código fonte bem estruturado e de fácil leitura. Não é a linguagem indicada para os principiantes, mas sobretudo para os programadores experimentados.

Também FORTH se encontra em posição intermédia entre o assembler e as linguagens de alto nível. Criado em meados dos anos setenta por Charles Moore, utilizou-se inicialmente para controlo em tempo-real de telescópios astronómicos. A sua principal característica é a de ser extensível: o programador pode definir novas funções e comandos que passam a fazer parte da linguagem.

Existem muitas outras linguagens de programação (mais de uma centena) pelo que a sua enumeração alongaria demasiado este artigo. Contudo, analizaremos em profundidade nas páginas que se seguem duas das mais utilizadas hoje em dia nos microcomputadores: Pascal e C.

A LINGUAGEM PASCAL

O Pascal é uma linguagem compilada. Isto quer dizer, que uma vez escrito o código "fonte", este se submete a um processo de tradução (compilação), cujo resultado é o código objecto ou programa executável, que normalmente se executa a partir do Sistema Operativo como mais um programa de serviço incorporado; para executá-lo, bastará escrever o seu nome, tal como acontece com qualquer outro comando do próprio Sistema.

DEVIDO ao seu caractér estruturado, e às suas possibilidades de cálculo científico e como é uma linguagem de alto nível, é muito utilizada nas escolas superiores de engenharia para o ensino de programação.

ESTRUTURA DE UM PROGRAMA

Um programa em PASCAL é composto de três partes fundamentais: cabeçalho, bloco declarativo e corpo do programa. O cabeçalho é constituído pela palavra PROGRAM e um nome para o programa. Segundo o PASCAL standard, depois do nome devem vir, entre parêntesis, as declarações dos ficheiros de entrada e de saída que o programa utiliza, embora alguns compiladores não necessitem delas, assumindo por defeito o teclado como ficheiro de entrada e a consola ou o ecrã como ficheiro de saída.

O bloco declarativo constitui a declaração das variáveis e tipos que o programa irá utilizar, assim como a declaração das funções e dos procedimentos. Uma das características do PASCAL, como da maioria das linguagens compiladas, é obrigar a declarar previamente os nomes e os tipos das variáveis, procedimentos ou funções que se vão utilizar no programa.

O corpo do programa é composto pelas instruções PASCAL que executam o processo que se deseja, no qual se podem utilizar com toda a liberdade as variáveis, tipos, funções e procedimentos declarados anteriormente.

A declaração de funções e de procedimentos segue a mesma estrutura do programa, se bem que no cabeçalho não se utilize a palavra PROGRAM, a não ser FUNCTION ou PROCEDURE.

ESTRUTURAS DE CONTROLO

Tal como acontece noutras linguagens, o PASCAL oferece ao programador diversas instruções para controlar o fluxo do programa. São elas: REPEAT...UNTIL.., WHILE...DO..., FOR...TO...DO..., FOR... DOWNTO...DO, IF...THEN...ELSE..., CASE...OF E GOTO. Vamos analisá-las uma a uma.

Devido ao seu carácter estruturado e as suas potencialidades para cálculo científico, o PASCAL é muito utilizado no ensino de Engenharia.

REPEAT... UNTIL permite que se repita a execução das instruções entre ambas as palavras até que se cumpra a condição depois de UNTIL. Por exemplo:

```
x : = 0;
REPEAT
        WRITELN(x);
        x : = x+1
UNTIL x=21;W
```

A primeira instrução atribui o valor 0 à variável x. depois entrarmos no ciclo escrevendo o valor de x e incrementando x em uma unidade.

Este processo repete-se até que se cumpra a condição escrita depois de UNTIL, isto é, até que x seja 21. Uma característica importante para desenvolver saltos condicionais consite em que, qualquer que seja o salto condicional em UNTIL, as instruções entre REPEAT e UNTIL executam-se sempre pelo menos uma vez.

WHILE...DO tem uma função semelhante a REPEAT .... UNTIL, ainda que a condição de saída se escreva atrás de WHILE, verificando-se antes de executar, as instruções entre WHILE e DO. Por exemplo:

```
x : =0;
WHILE x<22 DO
BEGIN
        WRITELN(x);
        x : = x+1
END.
```

Produz o mesmo resultado que o exemplo de REPEAT... UNTIL, se bem que se altere a primeira linha por x : = 22 o salto condicional não se executa. Numa outra situação, se mudarmos no exemplo de REPEAT... UNTIL a primeira linha por x : =22, o salto condicional executa-se a primeira vez, já que nesse caso a condição apenas se verifica depois de incrementar uma unidade a x.

FOR ...TO...DO

Os que conhecem o Basic não terão qualquer problema em comprender esta instrução, pois ela é idêntica à que se utiliza nessa linguagem. Quando se repetem as instruções situadas depois do DO, então o valor da variavel de controlo (que deve ser obrigatóriamente do tipo inteiro), inicializada a partir da igualdade situada depois do FOR, não se deve sobrepor ao valor situado a seguir ao DO. Após cada execução de ciclo o valor da variável de controlo incrementa-se de uma unidade. Por exemplo: FOR x : = 0 TO 21 do WRITELN(x);

Repare-se que já não é necessária x : =x+1, já que a instrução

FOR...TO...DO executa isso automaticamente.

A diferença em relação ao que acontece com o Basic, é que não se dispõe do modificador STEP para alterar o incremento da variável de controlo. Em PASCAL este incremento é sempre de uma unidade.

FOR...DOWNTO...DO

Este é um caso similar à instrução do Basic FOR...TO...STEP -1, pois o valor da variável de controlo é decrementada de uma unidade depois da execução do salto condicional, terminando-se quando o valor da variável de controlo é menor que o limite estabelecido a seguir a DOWNTO. Por exemplo: FOR x : = 21 DOWNTO 0 DO WRITELN(21-x);

Tem a mesma função que o exemplo para FOR... TO...DO.

IF...THEN ...ELSE

Esta instrução é totalmente igual à utilizada no Basic. Se a condição situada á frente da instrução IF for verdadeira executam-se as instruções situadas a seguir ao THEN, no caso de não ser verdadeira, executam-se as que estão na sequência de de ELSE. A instrução ELSE pode ser omitida. Exemplos:

IF x = 0 THEN WRITELN (x vale 0); IF x = 0 THEN WRITELN ('x vale 0); ELSE WRITELN ('x não vale 0').

CASE ...OF

Com esta instrução, pode fazer-se algo de mais semelhante à instrução ON...GOSUB do Basic. Após a instrução CASE deve ir o nome de uma variável de tipo inteiro, e em função do seu valor executa-se uma das instruções da lista situada na sequência de OF. Após cada instrução (ou grupo de instruções) deve surgir o valor correspondente da citada variável:

```
WRITELN ('Adivinha o numero em
que estou a pensar');
READLN;
READ (x);
CASE x OF
        0: WRITELN ('Não é zero')

  1. Porquê a unidade?
  2 .WRITELN   ('Parabéns, era o
dois!')
  3. WRITELN ('Passaste')
  4. OTHERWISE WRITELN (Número
demasiado alto')
```

A instrução OTHERWISE não é standard, incorporando-se somente em algumas versões. Significa "em qualquer outro caso" e as instruções situadas á frente dela executam-se se o valor contido na variável não corresponde a nenhum dos previstos na lista. No PASCAL standard simplesmente não se faz nada, passando-se com normalidade à instrução seguinte. De qualquer forma, quando se pretende incluir a mesma função que OTHERWISE em PASCAL standard, pode-se fazê-lo com a instrução IF. Por exemplo:

```
  WRITELN ('Adivinha o numero  em
que estou a pensar')
READLN ( adivinha o número em;
READ (x);
IF x > 3 THEN WRITELN ('Numero
demsiado alto')
 ELSE CASE % OF
   0: WRITELN ( Não é zero)
   1: WRITELN  (Porquê um UM)
   2: WRITELN ((Parabens, era o dois)
   3. WRITELN (' Já ultrapassaram'"
```

GOTO

A instrução GOTO é o "pato feio" do PASCAL, já que se recomenda que não seja usada, já que prejudica a clareza da estrutura do programa. Contudo, a instrução GOTO em PASCAL não é igual à do BASIC, já que em PASCAL não são utilizados números de linha. Como destino, utiliza-se uma etiqueta que é formada por um máximo de quatro dígitos e um mínimo de um e seguida por dois pontos. A etiqueta só se insere tal e qual no programa á frente da primeira instrução correspondente, mas é obrigatório declará-la na parte declarativa com a palavra reservada LABEL. Por exemplo:

```
  PROGRAMA (emprego (input, out-
put);
  VAR x:INTEGER;
  LABEL 10
  BEGIN
  x: =0;
  10:WRITELN(x);
  x:=x+1;
  IF x<22 THEN GOTO 10:
  END
```

FUNÇÕES

Em Pascal dispomos de diversas funções matemáticas e lógicas incorporadas, mas, por outro lado, o utilizador pode definir as suas próprias funções. As que estão incorporadas são as seguintes:

ABS(x) x deve ter um número ou variável inteira ou real . A função calcula o seu valor absoluto, que é o seu mesmo valor prescindindo do sinal.

SQR(x) x deve ser um número ou variável inteira ou real e a função calcula o quadrado do número.

TRUNC(x) x deve ser um número real, e a função calcula a sua parte inteira.

ROUND(x) x deve ser um número real e a função calcula o resultado arrendondando-o ao inteiro mais próximo por defeito ou por excesso. Se a parte fraccionária do x é exactamente 0.5, arredonda-se por excesso se o x é positivo e por defeito se o x é negativo.

SIN(x) x deve ser um número real e a função calcula o seno de x.

COS(x) x deve ser um número real e a função calcula o coseno de x.

ARCTAN(x) x deve ser um número real e a função calcula a arcotangente de x.

LN(x) x deve ser um número real e a função calcula o logaritmo natural de x.

EXP(x) x deve ser um número real e a função calcula o expoente de x.

SQRT(x) x deve ser um número real e a função calcula a raiz quadrada de x.

ORD(x) x deve ser do tipo CHAR (caracter) e a função devolve-nos o código ASCII do caracter x.

CHR(x) x deve ser um número inteiro e a função devolve o caracter que corresponde ao código ASCII x.

PRED (x) x deve ser do tipo CHAR e devolve o caracter anterior segundo a ordem do seu código ASCII.

SUCC(x) x deve ser do tipo CHAR e devolve o caracter seguinte segundo a ordem do seu código ASCII.

x AND y x e y devem ser variáveis lógicas, e a função é E lógico.

x OR y x e y devem ser variáveis lógicas e a função é OU lógico.

ODD(x) x deve ser um número inteiro e a função considera TRUE (verdadeiro) se o número x é impar e FALSE (Falso) se é par.

EOLN Esta função considera o valor como TRUE (verdadeiro) se foi alcançado o final de uma linha numa instrução de READ (leitura). Caso contrário considera FALSE (falso).

EOF Esta função devolve o valor TRUE (verdadeiro) se foi alcançado o final de um ficheiro antes de utilizar um procedimento GET ou depois de se utilizar o procedimento READ.

Ao dispor destas funções, o programador pode criar as suas próprias, com o número de parâmetros que desejar. Esses parâmetros podem ser valores, variáveis ou ponteiros de variáveis.

Os ponteiros são um tipo especial de variável própria de linguagens como o PASCAL e o C, sendo o seu conteúdo a direcção de memória em que se encontra outra variável. Deste modo, podemos utilizar ponteiros para números inteiros e para números reais, ponteiros para reais, ponteiros para booleanos e ponteiros de ponteiros. O uso de um

ponteiro variável como parâmetro de uma função permite-lhe atribuir valores ao procedimento que o chamou, assim como passar estruturas complexas (ARRAYS ou matrizes) como parâmetros. Se assim não fosse, uma função só poderia considerar um valor, o associado ao seu nome. Vejamos um exemplo simples de função:

```
FUNCTION media (a, b:real) : real;
BEGIN
   media: = (a+b)/2.0
END;
```

Como se pode ver, o nome da função é media e considerará um número real. Os parâmetros a e b também deverão ser números reais, e a única linha da função associa o seu nome (media) ao valor da semisoma de a e b. Para chamar a função basta utilizá-la como se se tratasse de uma das funções predefinidas:

```
m:=media (4,6);
```

Neste caso associa-se à variável real m o valor da media 4 e 6, isto é, 5.

UM PROGRAMA PASCAL CONSTA DE TRÊS PARTES FUNDAMENTAIS: CABEÇALHO, BLOCO DECLARATIVO E CORPO DO PROGRAMA

PROCEDIMENTOS

Para além das funções, o programador em PASCAL pode definir procedimentos. É um conceito similar às subrotinas do BASIC, mas com uma vantagem de que não se chama por número de linha, mas sim pelo seu nome, o que torna os programas mais legíveis. Por outro lado, como acontece na maioria das linguagens compiladas, o programador pode agrupar os procedimentos e funções mais utilizados por ele, uma vez testados e corrigidos os erros, nos ficheiros chamados *bibliotecas,* que lhe permitem aceder a elas a partir de qualquer programa que tenha sido criado, sem necessidade de reescrever o código fonte.

De uma forma mais simples, pode dizer-se que a principal diferença entre as funções e os procedimentos é que estes não podem atribuir algum valor associado ao seu nome, pelo que se um procedimento atribuir algum valor, tem de fazê-lo por referência (utilizando ponteiros como variáveis como foi referido antes).

Veja-se um exemplo simples:

```
PROCEDURE altera(VAR a, b:real);
VAR temp:real
BEGIN
   temp:=a;
   a:=b;
   b:=temp
END;
```

Como se pode ver, não se define nenhum tipo de variável para mudar, já que não se pode associar qualquer valor. Por outro lado, repare-se na palavra VAR a seguir aos parâmetros a e b. Isto faz com que o PASCAL durante a execução do programa, passe ao procedimento os endereços das variáveis a e b, de forma que as alterações produzidas pelo procedimento sejam permanentes ao voltar ao programa principal.

ESTRUTURA DE DADOS

O PASCAL utiliza cinco tipos de dados básicos: números inteiros (INTEGER), números reais (REAL), caracteres (CHAR), lógicos (BOOLEAN) e ponteiros. Estes ponteiros podem ser a qualquer dos cinco tipos, incluido ponteiros de ponteiros. Os quatro primeiros que referimos são escalares, isto é, compõem-se de um conjunto diferenciado de valores ordenados. Isto quer dizer que para qualquer par de valores a e b se dá uma destas três relações:

```
a>b
a=b
a<b
```

Deste modo, todos os operadores relacionais (>, >=, <,<=, = e <>) se podem aplicar a esses quatro tipos de dados escalares, sendo suposto que os dois operandos são do mesmo tipo.

Contudo, o PASCAL permite ao programador definir os seus próprios tipos de dados, o que permite facilitar a legibilidade dos programas e, por outro lado, permite criar tipos restritos. Por exemplo:

```
TYPE dia = integer
TYPE cor = (vermelho, verde, azul);
```

No primeiro caso, faz-se simplesmente que os dados tipo dia sejam inteiros, No segundo caso, cria-se um tipo novo, cor, que é formado somente por três elementos (vermelho, verde, azul) que cumprem a relação:

```
vermelho<verde<azul
```

Se no caso da definição do dia como inteiro utilizamos uma variável d para conter o dia da semana:

```
VAR d:dia;
```

ainda teremos que ter em conta que o valor da variável d está compreendida entre 1 e 7. Contudo, pode considerar-se o tipo dia de outra forma:

TYPE dia = (segunda-feira, terça-feira, quarta-feira, quinta-feira, sexta-feira, sábado e domingo);

então é o próprio compilador que testa o código fonte de modo que não seja atribuido um valor à variável d que não seja um dos citados na definição do tipo.

O PROGRAMADOR PODE AGRUPAR OS PROCEDIMENTOS E FUNÇÕES MAIS UTILIZADAS NAS BIBLIOTECAS, POUPANDO TEMPO NA FASE DO DESENVOLVIMENTO

O PASCAL permite também criar tipos que são subconjuntos de outros previamente definidos, pelo simples método de mencionar o primeiro elemento e o último do subconjunto. Por exemplo, uma vez definido o tipo de dia pode definir-se um tipo dia útil deste modo:

TYPE dia útil = segundas-feiras...sextas-feiras;

As reticências indicam que se trata de todos os dias da semana entre segunda e sexta-feira. No caso dos subconjuntos a declaração pode não fazer-se com TYPE, senão de uma forma implícita ao declarar a(s) variável(eis).

Passando a algo de mais complexo. O PASCAL permite definir as estruturas de dados conhecidas pelo número de vectores (ou matrizes, ou arrays), registos (record), ficheiros (file), conjuntos (set) e listas (list). Estas estruturas de dados são coleccções de dados organizados de uma determinada forma e construídas a partir de tipos elementares vistos anteriormente.

ARRAYS

É a estrutura mais simples do PASCAL, totalmente análoga às matrizes do BASIC. Trata-se de uma colecção de dados aos quais se acede através de um subíndice ou número de ordem, formando assim arrays unidimensionais. Podem ter-se também arrays de arrays, o que dá lugar a arrays multidimensionais (mais de um subíndice). Vejamos alguns exemplos:

```
VAR pontuação: ARRAY(1...18) OF
INTEGER;
VAR barcos:ARRAY(1...10)OF
ARRAY(1...10) OF BOOLEAN;
```

O tipo de barcos também poderia definir-se de uma forma simplificada como:

```
VAR barcos:ARRAY(1..10,1...10OF)
OF BOOLEAN;
```

Para aceder a um elemento de um array, bem como escrever um dado nele ou para ler o seu conteúdo, utiliza-se o/os subíndices. Por exemplo:

```
pontuação (3):=pontuação(3)+2;
IF NOT (barcos[3,6] THEN
```

WRITELN('!AGUA!') ELSE WRITELN('!IMPACTO!');

RECORDS

Os registos de PASCAL são estruturas de memória similares aos registos dos ficheiros de disco, já que estavam compostos por campos. Ver-se-á melhor com um exemplo de definição de um tipo de registo:

```
TYPE vendas=RECORD
        nome:ARRRAY(1...30) OF
CHAR;
direcção:ARRAY(1...80)OF CHAR;
telefone:ARRAY (1...15) OF CHAR;
        número:1..999;
        vendas:real
END;
```

Uma vez definido o tipo podemos usá-lo para definir variáveis:

VAR comercio:vendas;

e para aceder a um campo concreto utiliza-se o nome da variável separado por um ponto do nome do campo. Por exemplo:

```
comercio.telefone:='(93).7822310';
comercio.número:=34;
comercio.vendas:=2008930.00;
```

FILES

Os ficheiros em PASCAL são módulos de informação armazenados num suporte externo (disco magnético, fita magnética, fita perfurada, etc.) aos quais se chamou um nome. Ainda que normalmente os sistemas operativos ofereçam diversos meios para aceder aos dados de um ficheiro (sequencial, aleatório, etc.) o PASCAL só permite o acesso sequencial, isto é, para ler um determinado dado é obrigatório ler antes todos os anteriores.

Um ficheiro em PASCAL é uma colecção de unidades ou blocos com a mesma estrutura. Por exemplo, um arquivo pode estar composto de caracteres, arrays, registos, etc.., mas nunca de uma mescla deles. O final do ficheiro é detectado por uma marca especial ou marca de fim de ficheiro (EOF:End Of File marker).

O PASCAL proporciona ao programador um conjunto de instruções para criar, abrir, fechar, ler e escrever ficheiros, que são os seguintes: REWRITE, PUT, WRITE, RESET, GET, READ, EOF e EOLN.

REWRITE(f) é a instrução que permite inicializar (ou abrir) um ficheiro f para escrita. Se o ficheiro já existia, perde-se e começa de novo, No caso de não existir, é criado um novo ficheiro.

PUT(f) acrescenta o valor de fÎ ao arquivo f. O valor de fÎ fica indefinido até realizar a operação. PUT escreve sempre no final do ficheiro. A condição de final de ficheiro, EOF, há-de ser certa antes de usar PUT.

Há a salientar que as variáveis do tipo de ficheiro são implicitamente ponteiros, ainda que não se declarem como tal. Daí que no parágrafo anterior se utilize a expressão fÎ, a qual se refere ao dado (caracter, byte, inteiro, registo, etc.) ao que aponta f.

WRITE(f,x) atribui o elemento x a fÎ e executa a instrução PUT(f). WRITE só se aplica para ficheiros de texto.

RESET(f) inicializa (ou abre) um ficheiro f para leitura, colocando um ponteiro em relação ao primeiro elemento do arquivo (fÎ:elemento_1).

GET(f) coloca o ponteiro em relação ao elemento seguinte de f (fÎ:=elemento_seguinte). A função EOF(f) há-de ter um valor FALSE antes de efectuar um GET.

READ(f,e) equivale a e:=fÎ; GET(f);. Só é válido para ficheiros de texto.

EOF(f) é usado para verificar se fÎ está antes da marca EOF, em cujo caso EOF(f) é verdadeiro (TRUE).

EOLN está definida somente para os ficheiros de texto. O caracter de fim de linha é o retorno de cursor (CR). A função EOLN toma o valor TRUE antes da leitura de um caracter depois de CR. Se EOF (f) toma o valor TRUE durante a leitura do arquivo de texto EOLN(f), também tomará o valor verdadeiro.

A escrita em ecrã e leitura de teclado são formas especiais de texto para PASCAL, pelo que se utilizam instruções muito similares. Concretamente, READLN e READ para ler o teclado, e WRITELN e WRITE para escrever no ecrã. Para mais a função EOLN também está definida para leitura do teclado, de forma análoga ao manuseamento de ficheiros. Por outro lado, WRITE e WRITELN possuem modificações especiais que permitem formatar a apresentação dos textos e números do ecrã. Por exemplo: WRITELN (1234.567:10:2);

indica que há que utilizar dez dígitos (incluindo o ponto decimal) para escrever o número a que serão dados somente dois dígitos decimais. Como o número consta unicamente de oito dígitos, desvia-se para a esquerda com espaços. Assim, ao executar essa instrução no ecrã obteremos###1234.56, em que os símbolos # representam espaços em branco.

SETS

As estruturas SET ou conjuntos são uma colecção de objectos do mesmo tipo sem nenhuma ordem entre eles. O número máximo de elementos depende do compilador que está a ser utilizado. O tipo de objectos que pertence ao conjunto denomina-se tipo base do conjunto e pode ser qulquer tipo escalar, mas não um tipo estruturado (não podemos criar conjuntos de registos).

Com os conjuntos pode utilizar-se os operadores relacionais, mas com um sentido diferente:

= igualdade de conjuntos
<> desigualdade de conjuntos
<= inclusão (contido em)
>= inclusão (contém a).

Outros operadores para conjuntos são +(união), *(intersecção), - (diferença simétrica) e IN (inclusão).

Os três primeiros têm como resultado outro conjunto, enquanto que IN atribui um sinal booleano (TRUE ou FALSE).

Vejamos um exemplo de SET:

```
TYPE cores= (anil, azul, verde, amarelo, laranja, roxo, violeta, branco, negro)
VAR primárias: set of cores;
VAR secundárias: SET OF cores;
BEGIN
        primárias:=(azul,amarelo,rôxo);
        secundárias:=(anil, verde,
        laranja, violeta);
```

LISTAS

As listas(LIS) do PASCAL são estruturas de dados dinâmicas, já que o seu tamanho muda durante a execução. Uma lista é uma colecção de elementos do mesmo tipo, dispostos por ordem arbitrária ou ordenados. Cada elemento da lista tem um predecessor e um sucessor. O predecessor do primeiro elemento é um elemento vazio. O sucessor do último elemento é igualmente um elemento vazio. As listas caracterizam-se por ter as seguintes propriedades:

1º A maneira como se encadeiam entre si os elementos. Para tal fim utiliza-se normalmente o ponteiro.

2º A maneira como os elementos se podem inserir ou anular da lista. Estas alterações podem realizar-se em qualquer lugar da lista. Contudo, há listas do tipo pilha (stack) que só permitem alterações nos elementos inseridos mais recentemente.

3º O número de sucessores ou predecessores que cada elemento tem. Para a maioria das listas, só há um. No entanto, as listas tipo árvore binário admitem sucessores ou ramos por cada elemento (o ramo direito e o ramo esquerdo) assim como um "pai" ou predecessor.

O modo mais usual de utilizar as listas é com estruturas do tipo RECORD. Um exemplo simples e clássico seria este:

TYPE ponteiro= Îobjecto

```
objecto=RECORD
   seguinte:ponteiro;
   valor:tipodevalor
 END;
VAR p, base:ponteiro;
```

Deste modo em cada registo guardamos um dado (valor) e um ponteiro do registo seguinte.

Para gerir listas, o PASCAL possui diversos procedimentos que actuam directamente sobre ponteiros e que são NEW e DISPOSE.

NEW opera com um argumento do tipo ponteiro e cria uma variável dinâmica do mesmo tipo da variável que apontava o ponteiro. Uma vez criada a variável dinâmica há que atribuir-lhe um valor, tanto ao campo valor como ao campo seguinte, segundo uma estrutura que foi definida atrás:

```
base=NIL;
NEW(p);
pÎ.valor:=24;
pÎ.seguinte;base;
base:=p;
```

NIL é uma constante pré-definida pelo sistema, cujo valor se usa para indicar o valor do ponteiro situado no final da lista.

DISPOSE deixa livre a área de memória ocupada por um ponteiro. Este procedimento nem sempre se encontra no compilador, pelo que em determinadas ocasiões o programador deverá preocupar-se em recuperar no seu programa o espaço libertado. Determinadas implementações incluem procedimentos não mencionados aqui.

O PASCAL E A AMSTRAD

Para os computadores AMSTRAD nas suas diferentes gamas existem diversos compiladores de PASCAL. Para os AMSTRAD CPC está disponível o compilador da Hisoft, que gera ficheiros executáveis sob AMSDOS com extensão ".BIN", e incorpora diversos procedimentos e funções standard que permitem aproveitar os recursos do sistema operativo no que se refere a sons, gráficos, interrupções, etc.. Por outro lado, tanto para o CPC como para o PCW existe o Turbo Pascal (sobre CP/M Plus), muito interessante para os estudantes por ser uma versão do Turbo Pascal utilizado em algumas escolas de engenharia, e Pascal MT+, da Digital Research, muito interessante para os programadores, já que lhes coloca à disposição uma série de ficheiros fonte do assembler das rotinas principais de entrada/saída da biblioteca do sistema, o que lhes permite adaptá-lo a outros sistemas operativos (funciona sobre CP/M 2.2 e Plus).

Para a gama de computadores PC contamos com o poderoso Turbo Pascal, rápido em relação a outros compiladores, e com grandes facilidades de acesso às capacidades do computador (interupções,etc.), necessitando de pouca memória RAM e funcionando com aparelhos de uma única unidade de disco (nem todos podem dizer o mesmo). Também está disponível o compilador da Microsoft, o PASCAL COMPILER da IBM e o UCSD Pascal PSYSTEM de Intertec, entre outros.

EXEMPLO: TESTE COMPARATIVO (PASCAL, BASIC MALLARD E C)

Ainda que não seja possível incluir um teste comparativo a fundo de diversas linguagens, eis alguns pequenos exemplos. A listagem 1 está escrita em BASIC e trata-se de um teste típico de velocidade em operações aritméticas. Executado no BASIC Mallard do PCW, leva cerca de 252 segundos, e se for executado com um compilador de Hisoft num PCW, leva 35 segundos (listagem 2). E, por último, a listagem 3 realiza a mesma tarefa em C, e executado com o compilador da Hisoft leva somente 31 segundos.

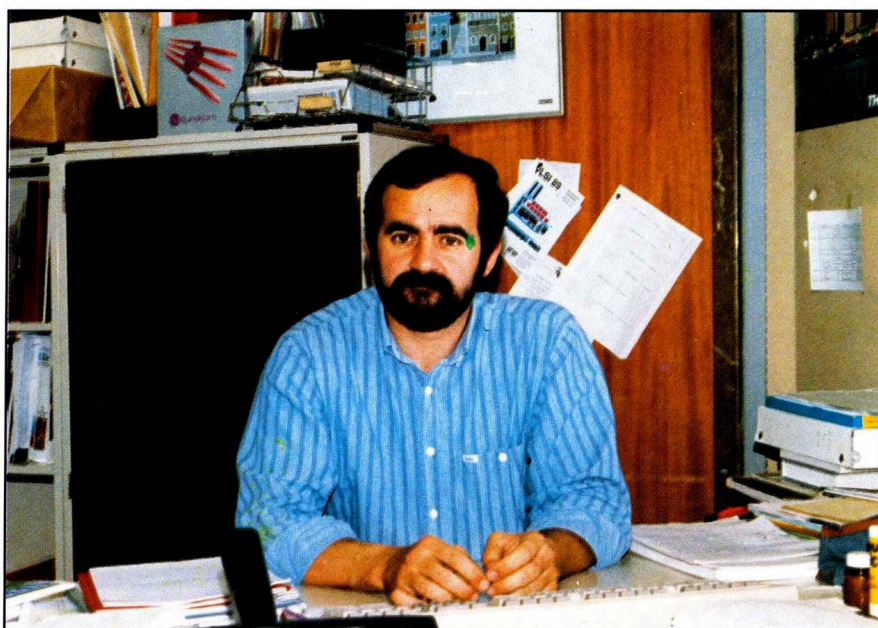
```
10 PRINT "Inicio..."
20 j%=7
30 FOR i=1 TO 30000
40 j%=j%+100
50 j%=j%-100
60 j%=j%*25
70 j%=j%/25
80 NEXT i
90 PRINT "... e fim"
100 END
```

```
PROGRAM aritmetica;

VAR i,j:INTEGER;

BEGIN
     WRITELN('Inicio...');
     j:=7;
     FOR i:=1 to 30000 DO
     BEGIN
          j:=j+100;
          j:=j-100;
          j:=j*25;
          j:=J DIV 25
     END;
     WRITELN('... e fim')
END.
```

```
main()

{ int i,j;

     printf("Inicio...\n");
     j=7;
     for(i=1;1<30000;++i)
     {    j=j+100;
          j=j-100;
          j=j*25;
          j=j/25;
     }
     printf("... e fim"\n);

}
```

O PASCAL É A ALTERNATIV PARA APRENDER A PROGR

— Defende o Prof. Pedro Veiga

Para um possuidor de um computador pessoal, para além da utilização de packages de processamento texto, folhas de cálculo, bases de dados, surge por vezes a necessidade de desenvolver aplicações numa linguagem de programação. Que linguagem escolher? Para esclarecer esta dúvida (e muitas outras) AMSTRAD MAGAZINE entrevistou o Prof.Pedro Veiga, docente do Instituto Superior Técnico e investigador do INESC.

A.M. - Quais são as linguagens de programação que considera terem marcado a história da informática?

Prof. Pedro Veiga - Antes de mais considero que será importante fazer um pouco de história da evolução das linguagens de programação. Quando os primeiros computadores começaram a aparecer a nível comercial, a sua programação era feita a um baixo nível. Eram programados em linguagem máquina, que é extremamente difícil de trabalhar porque obrigava a que existissem pessoas altamente especializadas, que mesmo assim acabavam por cometer muitos erros. Deu-se então um conjunto de passos com um objectivo: fazer linguagens de programação que fossem mais próximas da linguagem humana. A primeira grande tentativa foi criar a linguagem máquina simbólica, isto é, saindo da programação numa base binária (base 2) a qual ainda tinha um grande paralelo com a linguagem máquina. Nasce assim a linguagem Assembler que foi muito utilizada e ainda o é em duas áreas: quando não há possibilidade de recorrer a linguagens de mais alto nível (o que começa a ser cada vez mais raro), ou quando se quer fazer tarefas extremamente especializadas e eficientes. Devo acrescentar que hoje já existem linguagens de programação que permitem quase a eficiência de execução da linguagem Assembler (caso da linguagem C, de que falarei mais adiante). Fora desses casos, é totalmente desaconselhável usar a linguagem Assembler, porque é extremamente difícil sabê-la bem, por ser extremamente dependente da máquina - cada gama de computador tem, em princípio, a sua linguagem específica. Começaram entretanto a aparecer iniciativas para substituir o Assembler. A primeira linguagem de programação depois do Assembler foi o FORTRAN.

A IDEAL AO BASIC
AMAR

Esta linguagem foi criada por um grupo dirigido por um técnico da IBM chamado John Mackus. Depois de um processo mais ou menos longo para desenvolvimento, o FORTRAN foi a primeira linguagem a ser desenvolvida de um ponto de vista prático. O nome vem de FORmula TRANSlation porque se pretendia programar o computador sobretudo com base em fórmulas. No fim da década de 50, quando do desenvolvimento do FORTRAN, as principais aplicações eram de cariz matemático e destinavam-se ao campo científico ou militar.

A.M. - Entretanto houve necessidade de criar linguagens dirigidas à área comercial..

P.V. - De facto nos finais da década de 60 as aplicações comerciais começavam a ser importantes, aparecendo então a linguagem COBOL. Trata-se de uma linguagem radicalmente diferente do FORTRAN. O COBOL, na altura do seu lançamento, dizia-se que era para ser utilizada por qualquer pessoa. Hoje sabe-se que não é bem assim, porque também implica conhecimentos especializados.

Posteriormente foram desenvolvidas muitas linguagens, de entre as quais salientaria as que são os marcos fundamentais no desenvolvimento da informática. Temos assim o caso da linguagem ALGOL que já não é usada. Teve como ponto fundamental a introdução de muitos conceitos que ainda hoje são usados nas principais linguagens. Apareceria, entretanto, em 1964, a linguagem PL/1 da IBM.

Mas, a grande "pedrada no charco" surgiria com o nascimento da linguagem PASCAL. Teve dois aspectos fundamentais: primeiro, porque é extremamente simples do ponto de vista conceptual, tornando-a simples de aprender (PASCAL tem menos instruções que o BASIC) por novos utilizadores; por outro lado, introduziu conceitos fundamentais na estruturação de dados - coisa que não existia na maior parte das linguagens anteriores.

Em 1970, foi feito um inquérito e recencearam-se cerca de 200 linguagens de programação as quais, na sua maioria, eram usadas por grupos de utilizadores extremamente restritos. A evolução continuou e o PASCAL viria a ser melhorado pelo seu autor, Prof. Wirth, tendo dado origem entretanto à linguagem MODULA. Por sua vez, as linguagens "tradicionais" também sofreram algumas alterações e sobretudo melhorias, tendo sido introduzidos aspectos que eram considerados como bons para uma linguagem de programação: ser conceptualmente simples, ter instruções ou estruturas de controlo sintaticamente e semanticamente elegantes. Ou seja, não ter construções que do ponto de vista do utilizador fossem muito complicadas em termos de escrita das instruções, nem ser necessário entender o que é que o computador faz. Este ponto de vista de "entender" é importante porque há muitas instruções em linguagens de programação que são ambíguas, chegando a haver situações totalmente indesejáveis que são executadas de maneira diferente em computadores diversos. Uma outra característica fundamental nas linguagens de programação é uma adequada estruturação dos dados. Por exemplo, em linguagem BASIC se se pretender representar informação sobre uma pessoa - por exemplo, o seu nome, a sua idade, a altura, a cor dos olhos, etc. é necessário ter uma variável para cada uma dessas entidades. Em PASCAL pode criar-se uma superestrutura de dados em que uma só variável guarda toda essa informação. Esse facto é extremamente vantajoso porque permite fazer programas mais claros, menos susceptíveis de erro, o que permite aumentar a qualidade da programação.

Por outro lado, os conhecimentos de como é que as linguagens são processadas -basicamente, técnicas de construção de compiladores - tem trazido outros conhecimentos para melhorar as linguagens, de forma a que sejam o mais regulares possível para poderem ser processadas por compiladores muito eficientes, assim como outras "ferramentas" que o programador utiliza, como sejam "debbuger's" para auxiliar a produção de mecanismos de teste, etc.. Em suma, foram estes dois vectores - características das linguagens e características dos sistemas onde elas estão incluidas - que têm tido mais impacto sobre a criação das linguagens actuais.

DEPOIS DE SE SABER PASCAL É NECESSÁRIO CAMINHAR PARA LINGUAGENS MAIS AVANÇADAS

A linguagem PASCAL ainda tem uma grande importância hoje em dia, não tanto a nível comercial, mas sobretudo a nível universitário e, essencialmente, como primeira linguagem de ensino de programação! A partir daí, o PASCAL tem algumas desvantagens que desaconselham a sua utilização, nomeadamente, a indefinição das características de modularização - como é que vários módulos podem ser construídos por pessoas diferentes e integrados num só programa. Depois de se saber PASCAL é conveniente, ou torna-se necessário, caminhar para linguagens mais avançadas.

A.M. - As linguagens C e ADA têm sido muito faladas ultimamente. Poderão ser consideradas as linguagens do futuro?

P.V - A linguagem ADA foi criada por encomenda do departamento de Defesa dos EUA - um dos maiores centros mundiais de encomenda de software - para obviar ao grande problema da manutenção do software. Tinham programadores, uns especializados em COBOL outros em FORTRAN, etc. e a decisão foi tomada face à necessidade de programar aplicações extremamente complexas e com a necessidade de serem mantidas por muitos anos, que é algo que as pessoas que fazem programas por vezes não pensam. Os programas em muitos casos terão de funcionar durante cerca de 20 ou 30 anos -

casos típicos de tráfego aéreo, sistemas militares, sistemas bancários, etc. - que embora não se mantenham sempre intactos, vão sendo adaptados a novas necessidades e a novos equipamentos.

A linguagem ADA aproveitou muitos conhecimentos do PASCAL e está a ter muita implantação nos EUA. É difícil prever qual vai ser o impacto da linguagem ADA daqui por 10 anos. No entanto ela já tem em muitas áreas - sobretudo aplicações militares - uma grande implantação. É uma linguagem que da minha perspectiva tem algumas desvantagens. É extremamente "grande", sendo consequentemente difícil de aprender. Ora este aspecto de dificuldade de aprendizagem é uma desvantagem que as linguagens de programação não devem ter. Há ainda uma outra particularidade que tem a ver com o facto de ser pesada em termos de processamento sendo os compiladores pouco eficientes. De referir que ADA é uma linguagem destinada a programar todo o tipo de aplicações, nomeadamente as aplicações de tempo-real que são extremamente importantes hoje em dia.

A GRANDE DIFUSÃO DA LINGUAGEM C DEVE-SE À GRANDE EXPANSÃO DO SISTEMA UNIX

A outra linguagem que efectivamente já tem um grande impacto na actualidade e que se prevê que venha a aumentar, é a linguagem C. Foi desenvolvida no princípio da década de 70 nos laboratórios Bell, tendo sido especialmente concebida para reunir as seguintes características: ser uma linguagem de alto nível, isto é, fácil de entender pelos utilizadores; ser uma linguagem extremamente eficiente quando executada. A grande difusão da linguagem C deu-se quando o sistema operativo UNIX começou a ser divulgado, devido ao facto da linguagem C ter sido escolhida para escrever quase todo o sistema operativo UNIX. Ora, por se tratar de uma linguagem de alto nível tornou mais fácil a possibilidade de exportar os programas para outro computador, bastando para o efeito ter um compilador de C. Como já referi a C é uma linguagem extremamente eficiente quase ao nível do ASSEMBLER. Com a crescente difusão de sistemas UNIX a linguagem C está a ser extremamente popular.

A.M. - Na qualidade de docente do I.S.Técnico, como vê o ensino das linguagens de programação?

P.V. - Normalmente, os alunos aprendem a programar em PASCAL que, como já referi, reune um conjunto de características muito boas para a pessoa aprender programação numa primeira aproximação. Quando os alunos já dominam muito bem a programação, é-lhes ensinada a linguagem C e, a partir daí, passam a desenvolver aplicações. Aprender C como primeira linguagem é extremamente desaconselhável na medida em que é uma linguagem que tem características que não permitem ensinar de uma maneira clara alguns conceitos de programação.

APRENDER PROGRAMAÇÃO COM BASIC: "QUEM NASCE TORTO TARDE OU NUNCA SE ENDIREITA"

A.M. - Mas a linguagem BASIC tem sido para muitos a prmeira linguagem de programação...

P.V. - O BASIC foi criado em meados da década de sessenta, em primeiro lugar para ambientes interactivos e depois, foi apontado como sendo uma linguagem para principiantes. Trata-se realmente de uma linguagem extremamente simples de aprender - eu próprio já sabendo previamente de FORTRAN, aprendi BASIC em 15 minutos. É, em minha opinião, uma linguagem extremamente má. Digo isto por diversas razões: em primeiro lugar, por não existir um só BASIC. Existem muitas implementações de BASIC o que é extremamente mau para o programador do ponto de vista de aprendizagem e do ponto de vista dos programas que faz. Uma pessoa que faça um programa num computador da marca x se o levar para um Basic diferente tem de fazer muitas alterações. Em segundo lugar, a linguagem Basic não tem estruturação de dados conveniente. As estruturas de controlo (instruções) são extremamente deficientes e não tem características de modularização de programas e tem conceitos de sub-rotinas extremamente primitivos. Numa altura em que tivemos de ensinar BASIC no Técnico, os alunos tinham extrema dificuldade em fazer programas complicados, situação em que esta se torna praticamente inutilizável devido ao grau de esforço ser muito grande. Só um inconsciente é que hoje faz grandes aplicações em BASIC. Seria como fazer uma outra ponte de Lisboa à outra margem do Tejo em bocados de madeira...

Por outro lado, da experiência que tive a dar aulas de programação de PASCAL no Técnico, havia dois grandes grupos: os alunos que nunca tinham programado e os alunos que já sabiam fazer programas, nomeadamente em Basic. O que se constatava é que os trabalhos finais feitos pelos diversos alunos em média eram melhores do que aqueles que eram feitos pelos alunos que nunca tinham programado. A explicação tem a ver com o facto de que aqueles que já sabiam BASIC tinham certos vícios de programação, nomeadamente, por exemplo, na utilização de estruturas de dados.

Portanto, a linguagem BASIC tem, do ponto de vista de utilização para um iniciado em informática algumas vantagens: é simples. Mas, repito, tem muitas desvantagens que normalmente limitam a sua capacidade de evolução. Há um ditado português que diz: "o que nasce torto tarde ou nunca se endireita"...

A.M. - Entretanto, tem-se vindo a falar de linguagens da 4ª geração. Qual vai ser o seu papel no futuro da informática? Que importância têm as novas metodologias de programação?

P.V. - Esta evolução das linguagens teve em paralelo uma evolução das metodologias de programação. Durante muitos anos falou-se de programação estruturada - que era um conjunto de técnicas que permitia construir um programa de uma forma hierarquizada, dividindo as tarefas muito complexas em tarefas cada vez mais pequenas, e indo programar cada uma delas, numa estratégia de "dividir para conquistar". As linguagens à medida que foram evoluindo foram se adaptando a isso. A linguagem PASCAL é extremamente adequada para fazer programação estruturada, enquanto que o BASIC não é. A vantagem de um programador utilizar uma linguagem estruturada é porque ela leva automaticamente a uma expressão em termos mais correctos e mais estruturados. Há portanto características inerentes às próprias características das linguagens que foram feitas para se adaptarem às metodologias de programação que entretanto tinham sido desenvolvidas. Claro que esta evolução das linguagens não é estática. Continuou-se a desenvolver mais linguagens, sendo a perspectiva actual criar mais linguagens que permitam ao programador tarefas de cada vez mais alto nível. Vejamos um exemplo: um programador tem uma variável onde guarda um tempo (horas, minutos

e segundos) e suponhamos que tem outra variável onde guarda tempo. Se se quiser somar as variáveis, o programador tem de saber que os segundos e os minutos são somados em módulo 60, enquanto que as horas são em modo 24. Isto programado numa linguagem normal, mesmo em PASCAL, tem de ser feito à custa de muitas instruções, para ir buscar os diversos componentes e somá-los consoante o significado da variável que lá está embebida. Hoje em dia começa-se a falar em linguagens mais avançadas que permitem um encapsulamento maior entre dados e as operações que são feitas sobre esses dados. Essas linguagens fazem parte de um conjunto geral que por questões de marketing se costuma chamar de linguagens da 4ª geração. Têm muitas facetas diferentes, mas têm a característica de poder levar o utilizador a programar a um nível mais elevado. Têm operações primitivas muito mais avançadas do que outras linguagens. Por exemplo, em Smalltalk, não existem operadores convencionais, existindo mecanismos de comunicação dentro da linguagem. Por exemplo, quando se pretende fazer uma operação entre duas variáveis, isso é feito criando uma mensagem específica em que vão os dois operandos e a mensagem é identificada como sendo de soma. Claro que este conceito pode ser estendido para operações muito mais complicadas que soma. Portanto, essas linguagens permitem trabalhar a um nível muito mais alto.

A nível do INESC nós usamos uma linguagem que é o C++ que é um passo de evolução da linguagem C na direcção das chamadas linguagens orientadas por objectos, que começam a ser as preferidas em alguns tipos de ambientes,permitindo ao programador fazer programas mais próximos da sua linguagem normal e, por outro lado, mais distantes do computador. Isso é vantajoso porque relegam para o computador as tarefas fastidiosas que são susceptíveis de erro. Assim, o programador produz programas mais compactos, mais legíveis.

A.M. - A linguagem APL é um caso à parte no panorama das linguagens de programação. Qual é campo de acção desta linguagem?

P.V. - Na verdade a APL (A Program Language) é uma linguagem que se distingue de todas as outras. Foi desenvolvida por um cientista da IBM para ambientes interactivos. Tem uma particularidade que tem limitado a sua difusão, que é o facto de necessitar de um conjunto de caracteres extremamente especial. Por exemplo, existem símbolos especiais para instruções de leitura , de escrita, para operações com matrizes, etc. O conjunto de caracteres é extremamente extenso, muito mais que os ASCII normal , o que obriga a usar teclado e monitores especiais. É, no entanto, uma linguagem extremamente elegante e permite fazer programas extremamente compactos e legíveis. Nos EUA é usada como primeira linguagem de ensino de programação. No I.S.Técnico teve um clube de fãs bastante grande. Acaba por ser uma linguagem com um campo de utilizações relativamente limitado.

A.M. - Quais são os objectivos das metodologias de programação?

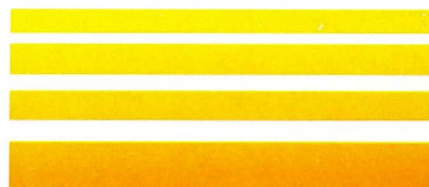
P.V. - Primeiro, permitir fazer programas extremamente resistentes a erros, isto é, que são concebidos a partir de especificações de tão alto nível quanto possível. As várias metologias têm como preocupação permitir estruturar etapas desde as especificações até ao código produzido de uma maneira o mais simples possível. Uma das preocupações que as metologias de programação têm procurado atingir é permitir facilitar as tarefas de teste. Trata-se uma tarefa extremamente complexa e reconhece-se hoje em dia que o problema de testar um programa pode ser isolado da sua concepção. Por exemplo, posso dizer que nos laboratórios e nas fábricas de software do mundo, existem normalmente duas equipas - uma que faz o software e outra que prepara os testes para esse software. A tarefa das metologias de programação é "casar" estas duas tarefas que são fundamentais o mais cedo possível. Um outro aspecto importante na produção de software diz respeito à manutenção, que deve ser entendida como uma tarefa para melhorar um produto para atender às novas especificações que naturalmente vão surgindo ao longo do tempo. Manter significa também corrigir os erros que vão aparecendo. Os erros que infelizmente aparecem podem surgir ao fim de muitos anos, relativos a uma especificação que não foi pensada na fase inicial.

Um outro aspecto importante actualmente tem de ter em conta que um produto de software não é feito por uma pessoa, mas sim por uma equipa de técnicos de programação. Em qualquer sitio que trabalhe uma equipa de pessoas é preciso organização e haver mecanismos de comunicação entre essas pessoas e mecanismos para garantir a integridade das várias parcelas do trabalho feito pelas diversas pessoas. Portanto, os ambientes de programação procuram actualmente facultar ao programador mecanismos de gestão de projectos que são feitos por várias pessoas. Por exemplo, no âmbito do programa europeu ESPRIT, foi criado um ambiente de programação PCTE - Portable Common Tool Environment - que é um conjunto de mecanismos e "ferramentas" que permitem criar sobre esse ambiente aquilo que eles designam uma fábrica de software, em que em cada instalação, o ambiente é adaptado ás suas necessidades. Este ambiente PCTE está a ter um grande impacto a nível europeu e gostariamos que em Portugal também começasse a ser mais divulgado, por ser excelente na coordenação de equipas numerosas. Existe no INESC uma instalação piloto.

A.M.- Concretamente, em relação às linguagens para MS-DOS, portanto aplicáveis para Computadores Pessoais, que linguagens aconselharia?

P.V. - A maioria das linguagens que referi atrás estão disponíveis no mercado - BASIC, FORTRAN, COBOL, PASCAL , C, alguns subconjuntos de ADA e ainda LISP e PROLOG. Das implementações que existem em PASCAL, a mais popular é o Turbo PASCAL que infelizmente tem algumas desvantagens, pois não segue totalmente a norma PASCAL . Tem alguns desvios em relação ao PASCAL normalizado, sendo um desvio em relação à norma sempre indesejável porque dificulta a tarefa do programador e o transporte da aplicação para outro computador. Por outro lado, o TurboPASCAL é um pouco fraco em termos de verificação dos programas, nomeadamente ao nível de aferição sintática e semântica.

A.M. Qual será então a linguagem ideal para aprender a programar?

P.V.- Entre o BASIC e o PASCAL não tenho dúvidas em sugerir esta última. Consequentemente desaconselho, pelas razões que foram enumeradas atrás, que alguém que queira aprender a programar o faça através do BASIC.

PS/2 E OS/2 - UM ANO DEPOIS

Foi exactamente há um ano que a IBM introduziu a sua linha de computadores pessoais PS/2 e o sistema operativo OS/2, desenvolvido em colaboração com a Microsoft. Este artigo pretende dar uma perspectiva actual do PS/2 e do OS/2 e o que estes produtos significam para os vários implicados na sua utilização.

MESMO após um ano existe ainda uma considerável confusão sobre o que é exactamente o PS/2 e o OS/2. Arriscando-nos a repetir factos já demasiado falados, vamos começar pelo pincípio.

CONSIDERAÇÕES GERAIS

PS/2 é o nome dado pela IBM ao que é actualmente (e ao contrário da impressão que causou o seu anúncio) a mais heterogénea linha de computadores pessoais. Os modelos 25 e 30 da gama PS/2, construídos sobre o processador Intel 8086, não utilizam o BUS MCA (Micro-Channel Architecture) ou o adaptador gráfico VGA (Video Graphics Array), existente nos modelos mais caros. Além disso, não podem correr o sistema operativo OS/2. Contudo, estes modelos atingiram 60% das entregas dos PS/2 da IBM em 1987.

O sistema operativo OS/2, desenvolvido conjuntamente pela IBM e a Microsoft, foi desenhado especificamente para máquinas baseadas nos processadores de 16 bits Intel 80286. Mas, com o mais poderoso processador desta marca (80386) o OS/2 pode trabalhar no modo 286, já que o trata como se fosse um 286 mais rápido.

Mas, mais ainda, todas as versões anunciadas do OS/2 têm limitações, ligadas directamente às próprias limitações de concepção dos processadores Intel 286. A mais significativa destas está relacionada com a possibilidade de correr aplicações DOS sob OS/2. A compatibilidade com o DOS requer a suspensão da multitarefa, pode criar problemas com outras aplicações a correr sob o OS/2 e não trabalha bem com tarefas sensíveis ao 'timing', tais como programas de comunicações.

Os genuínos PS/2, modelos 50 e 60, baseados nos microprocessadores 80286 a 10 Mhz, mais o modelo 80 (baseado no 80386 a 16 Mhz), constituiram 38% das entregas IBM em 1987. Parece, no entanto, que estes modelos continuam a ser minoritários nas vendas dos sistemas PS/2 nos primeiros meses de 1988. Esta minoria é a única capaz de correr o OS/2.

Ironicamente, ainda que imbuído das limitações do 80286 da Intel e não aproveitando a maior potência do 80386, o OS/2 não corre muito bem nos sistemas baseados no 286. A versão básica (Standard Edition Version 1.0) trabalha razoavelmente nos sistemas 286. Se este OS/2 é, contudo, apenas o esque-

leto do projecto OS/2 (proporcionando pouco mais que a multitarefa e o acesso aos 16 Mb de memória, endereçáveis pelo 286) tem o seu maior aperfeiçoamento com o interface gráfico incorporado no Windows Presentation Manager da Standard Edition 1.1, marcada para comercialização pela IBM para Outubro de 1988. É este interface que realmente abre a porta para um software mais poderoso e fácil-de-usar.

Existem, contudo, razões crescentes para crer que a Standard Edition 1.1 do OS/2 (com o Windows Presentation Manager) não irá oferecer performances satisfatórias em qualquer sistema baseado no 80286. Este processador não é suficientemente rápido para proporcionar um nível de resposta de sistema adequado à maioria dos utilizadores. Mesmo com o 286 a 12 Mhz corre muito lentamente. Os PS/2 50 e 60 baseados no 286 a 10 Mhz são ainda mais lentos. De facto, mesmo um sistema baseado no 80386 a 16 Mhz como o modelo 80 não vai ser sempre tão rápido como a maioria esperaria quando corre a versão completa do OS/2 Standard Edition 1.1. Um processador mais rápido que este seria, assim, desejável.

Os possuidores do OS/2 Extended Edition, quer na versão 1.0 sem o Presentation Manager, quer na versão 1.1 com ele, iriam requerer mais recursos (velocidade de processador, memória e capacidade de memória em disco) que os utentes da Standard Edition. O OS/2 EE versão 1.1 apresenta uma performance ainda menos satisfatória que a Standard Edition 1.1 nos sistemas baseados no 286. O resultado final é que torna-se provável que só uma muito pequena minoria dos originais PS/2, modelos 50 e 60, irão ser, na realidade, utilizados para correr totalmente o OS/2, quer na versão Standard quer na Extended. Pura e simplesmente estes modelos não são suficientemente poderosos para fornecer uma performance que satisfaça. De facto, apesar de vocacionado para o Intel 286, é altamente provável que o OS/2 não vá ter uma utilização significativa em sistemas baseados em versões mais rápidas deste processador- por exemplo, a 16 Mhz. É mais provável, contudo, que seja apenas nos sistemas baseados no 386, caso do PS/2 modelo 80, que o OS/2 venha a ganhar uma aceitação real.

Isto quererá, provavelmente, dizer que este sistema operativo não será usado correctamente antes da segunda metade de 1989, em que se imagina que uma versão posterior, específica para o 386, esteja disponível. Até lá não é provável que o OS/2 atinja um número real de utilizadores, pelo menos próximo do número que a imprensa pró-IBM poderá fazer crer.

Felizmente, embora exista alguma confusão, deve ser notado que, apesar do "/2" nos nomes, não há nenhuma relação particularmente estreita entre o PS/2 e o OS/2.

O PS/2 modelo 60 da IBM, que é o modelo que realmente deve ser considerado como correndo o OS/2 satisfatoriamente, representa pouco mais de 5% das vendas dos PS/2 da IBM em 1987.

A IBM introduziu recentemente modelos mais potentes do PS/2 — o modelo 50Z, com processador 286 a 10 Mhz, a série 70, três modelos com processador 386 a 16, 20 e 25 Mhz e ainda uma nova versão do modelo 80 a 20 Mhz. Não obstante ter sido grande a divulgação dada ao seu lançamento, o facto dos PS/2 da IBM não oferecerem um aumento de performance sobre o Standard Industrial AT Compatível significa que a IBM não tem vantagens reais sobre as outras marcas com o PS/2 e o OS/2.

QUE CONSEQUÊNCIAS PARA OS UTILIZADORES?

A consequência mais provável para os utilizadores é que um grande número de possuidores IBM PS/2 modelos 50 e 60 irão ficar significativamente desapontados. Mais ainda os que compraram sistemas baseados no 286 a 10 Mhz, acreditando que, através dele e com o OS/2, estavam a ganhar o acesso a uma nova geração de software para PC's. E mesmo que estejam dispostos a gastar o dinheiro extra requerido para a memória viva (o aconselhado é, pelo menos, de 1,5 a 2 Mb) e memória de massa necessária para usar realmente o OS/2 nos modelos 50 e 60, irão descobrir que o sistema não é, na realidade, suficientemente rápido para que possam fazer dele uma utilização para além do mínimo das capacidades deste sistema operativo.

A performance do presentation Manager do OS/2 assemelhar-se-á à do Windows 1.0 da Microsoft no velho PC XT. Trabalha, mas é demasiado lenta para ser utilizável.

O resultado deverá ser uma considerável desilusão também com o OS/2, que fica muito abaixo das promessas e promoções da IBM para ele apresentadas. Por outro lado, as vendas do Standard Industrial AT Compatível são ainda fortes e parecem continuar de boa saúde, em comparação com as vendas do PS/2.

Ninguém gosta de ser "levado". Infelizmente muitos compradores do PS/2 estão, ao fim de algum tempo, a sentir que foram levados pelos anúncios e pelos reclamos promocionais da IBM para o PS/2 e para o OS/2. Como consequência, eles irão optar, mais voltados para fontes alternativas em termos de PC's.

CONSEQUÊNCIAS PARA A IBM

Não é preciso ter muita imaginação para concluir que nenhuma destas notícias é boa para a IBM. E, embora muitos utilizadores ainda acreditem que a linha PS/2 tem qualidades mágicas, a IBM continua a perder mercado para os concorrentes. Agora que começa a ficar claro quantos fatos o imperador leva vestidos, as deserções para outras marcas parecem acelerar-se.

Tristemente, foram os fregueses mais leais e confiantes da IBM que pensaram que, comprando um PS/2 modelo 50, era uma forma de adquirirem a tecnologia do futuro. Compreenderam assim que apenas compraram uma versão não muito standard do que realmente é a tecnologia do passado. E não continuarão a ser confiantes e leais.

As pessoas da Entry Systems Division da IBM percebem muito bem, pensamos, a amplitude do problema que criaram para eles próprios e farão tudo o que puderem para o resolver. Existe uma grande preocupação para a introdução de sistemas, tanto quanto possível, mais rápidos.

A IBM está, neste momento, numa posição defensiva em relação aos computadores pessoais, encarando uma dura escolha entre aguentar a linha de preços e continuar a perder o mercado, ou reduzir os preços e perder as margens. O mais provável será esforçar-se para fazer um pouco de cada, sofrendo, contudo, a consequência de ambas as decisões.

CONSEQUÊNCIAS PARA OS 'DEALERS'

Embora se preveja um abrandamento nas vendas dos computadores pessoais em termos internacionais, é provável que este ano os 'dealers' do PC Standard baseado no BUS IBM PC/AT e extensões deste, continuem a ter melhores resultados que a IBM. A falta de qualquer vantagem prática no BUS de arquitectura Micro-Channel, combinado com outros problemas da IBM, assegura a estes vendedores a continuação de ganhos interessantes.

CONSEQUÊNCIAS PARA AS "SOFTWARE-HOUSES"

Talvez as consequências mais dramáticas do estado actual do mercado, no âmbito do PS/2 e do OS/2, sejam para as software-houses.

Se uma empresa contar apenas com as vendas de software concebido especificamente para o OS/2 e para o Presentation Manager, o melhor será ter os bolsos bem cheios. A espera até que o OS/2 SE 1.1 (com o Presentation Manager) fique disponível, e o pequeno número de sistemas que irão ser instalados, mesmo juntando os sistemas de outras marcas que forem capazes de funcionar com ele, são factos que acrescentam muito poucas oportunidades de fazer dinheiro vendendo software baseado no OS/2- pelo menos a breve prazo.

Na prática, não é provável que o software escrito na totalidade para o OS/2 venha a significar muito, quer em número de unidades vendidas, quer em valor, antes de meados de 1989, no mínimo.

Uma perspectiva mais conservadora (e talvez mais realista) é a de que se deve chegar a 1990 antes que alguém esteja realmente apto a esperar largas vendas com o software desenvolvido para o OS/2. Uma breve análise do software já existente para o OS/2 Standard Edition 1.0 (sem o Presentation Manager) mostra-se algo melhor em perspectivas, mesmo que ainda relativamente modesto. Na prática, contudo, muitos dealers de software verão que a capacidade de mercado é muito limitada. Muitas casas de software acabam simplesmente por manter as suas versões tradicionais sob PC/DOS, fornecendo-as em disquetes de 3,5 polegadas.

Na realidade, o software, para tirar vantagens do Presentation Manager, não terá níveis de vendas substanciais até que o OS/2 com o Presentation Manager esteja disponível e um grande número de sistemas capazes de funcionar com ele estejam instalados. Isto levará provavelmente, pelo menos, ainda um ou dois anos.

Significa isto que as software-houses que quiserem começar com produtos que, eventualmente, possam utilizar sobre o Presentation Manager, vão descobrir que o melhor é escrevê-los em primeiro lugar para o Microsoft Windows 2.0/286.

REDEFINIÇÃO DO TECLADO COM KEYS.WP

A face 1 das disquetes de sistema contém o ficheiro KEYS.WP que se utiliza em conjunto com o programa SETKEYS (o comando completo é SETKEYS KEYS.WP). O seu objectivo é definir o teclado para poder utilizar os programas (geralmente processadores de texto e editores) que operam com a norma de teclas WORDSTAR (entre outros, o editor de dBASE II guia-se por ele). Na realidade, não é imprescíndivel carregá-lo para que os ditos programas funcionem, já que a norma citada se baseia em combinações de teclas com CONTROL e ESC (ALT ou EXIT no PCW), mas serve para tornar o seu uso mais fácil e para aproveitar certas teclas especiais de que dispõe o nosso computador. Algumas teclas (como a seta do cursor) não realizarão a função que se espera delas, ainda que, como já dissemos, se possam substituir por outra combinação de teclas.

Por exemplo, para justificar um parágrafo que se "deslocou" ao inserir ou apagar texto, no WORDSTAR utiliza-se CONTROL (ou ALT) + B. No entanto, sempre será mais cómodo carregar simplesmente na tecla [RELAY], que se encontra no PCW, pelo que será mais útil redefinir a dita tecla com o valor CONTROL + B.

Na sequência damos uma listagem do ficheiro KEYS.WP com uma série de comentários. Desta maneira será muito mais fácil compreender o conteúdo deste ficheiro, saber as teclas que se devem digitar em cada ocasião, e observar a forma como trabalha o programa SETKEYS que permite definir o teclado.

Observe que se colocaram três colunas à direita: as duas primeiras são para expressar a combinação de teclas que se está a redefenir (em alguns casos definem-se duas combinações diferentes, noutros apenas uma) e a última fila da esquerda define o resultado que se consegue.

```
14 N S  "↑E"       ↑E     ↑              ↑ SHIFT            Linha para cima
14 A SA "↑'#9E'"   ↑QE    ↑ ALT          ↑ SHIFT+ALT        Parte superior do ecra
 6 N S  "↑D"       ↑D     ->             ->+SHIFT           Caracter para a direita
79 N S  "↑X"       ↑X     v              v+SHIFT            Linha para baixo
79 A SA "↑'#98'"   ↑QX    v+ALT          v+SHIFT+ALT        Parte inferior do ecra
15 N S  "↑S"       ↑S     <-             <-+SHIFT           Caracter para a esquerda
 5 N    "↑D"       ↑D     CHAR                              Caracter para a direita
 5 A    "↑S"       ↑S     CHAR+ALT                          Caracter para a esquerda
 5   S  "↑F"       ↑F     WORD+SHIFT                        Palavra para a direita
 5   SA "↑A"       ↑A     WORD+SHIFT+ALT                    Palavra para a esquerda
13 N    "↑'#9C'"   ↑QD    EOL                               Fim de linha
13   S  "↑'#9C'"   ↑QD    LINE+SHIFT                        Fim de linha
13   SA "↑'#9D'"   ↑QS    LINE+SHIFT+ALT                    Comeco de linha
12 N    "↑C"       ↑C     PAGE                              Pagina para baixo
12 A    "↑R"       ↑R     PAGE+ALT                          Pagina para cima
12   S  "↑'#90'"   ↑QC    DOC+SHIFT                         Comeco do ficheiro
12   SA "↑'#91'"   ↑QR    DOC+SHIFT+ALT                     Final do ficheiro
20 N    "↑'#92'"   ↑QF    FIND                              Procurar
20   S  "↑'#93'"   ↑QA    EXCH+SHIFT                        Alterar
10 N    "↑'#94'"   ↑KB    CUT                               Marcar principio do bloco
10   S  "↑'#95'"   ↑KK    CUT+SHIFT                         Marcar final do bloco
11 N S  "↑'#96'"   ↑KC    COPY           COPY+SHIFT         Copiar blocos
 3 N S  "↑'#97'"   ↑KV    PASTE          PASTE+SHIFT        Mover blocos
 1 N S  "↑B"       ↑B     RELAY          RELAY+SHIFT        Reformatar o paragrafo
23 N S  "↑V"       ↑V     [+]            [+]+SHIFT          Inserir Sim/Nao
16 N S  "↑G"       ↑G     DEL->          DEL->+SHIFT        Apagar este caracter
16 A SA "↑'#9A'"   ↑QY    DEL->+ALT      DEL->+SHIFT+ALT    Apagar ate ao final da linha
72 A SA "↑'#9B'"   ↑Qdel  <-DEL+ALT      <-DEL+SHIFT+ALT    Apagar ate ao comeco da linha
 8 N S  "↑'#1B'"   esc    EXIT           EXIT+SHIFT         ESC
66 N S  "↑U"       ↑U     STOP           STOP+SHIFT         Interromper o processo

E #90 "↑QC"
E #91 "↑QR"
E #92 "↑QF"
E #93 "↑QA"
E #94 "↑KB"
E #95 "↑KK"
E #96 "↑KC"
E #97 "↑KV"
E #98 "↑QX"
E #9A "↑QY"
E #9B "↑Q↑'#7F'"
E #9C "↑QD"
E #9D "↑QS"
E #9E "↑QE"
```

Listagem comentada do ficheiro KEYS.WP

AGENDA

DATA	CIDADE (PAÍS)	CERTAME
12/9 a 16/9	Nice (França)	EUROGRAPHICS 88
13/9 a 16/9	Paris (França)	OEM 88
14/9 a 19/9	Milão (Itália)	SMAU e INTERNATIONAL OFFICE EXHIBITION
14/9 a 18/9	Londres (Inglaterra)	PERSONAL COMPUTER SHOW
19/9 a 24/9	Paris (França)	SICOB MICRO
13/10 a 16/10	Porto (Portugal)	ESCRITÓRIO
20/10 a 25/10	Colónia (RFA)	ORGATECHNICK
25/10 A 28/10	Munique (RFA)	SYSTEC
27/10 a 30/10	Porto (Portugal)	INFORPOR
14/11 a 18/11	Las vegas (USA)	COMDEX FALL
22/11 a 25/11	Lisboa (Portugal)	ENIC

CONCURSO

MELHOR PROGRAMA (SÓ ATÉ 30 DE SETEMBRO)

LEITORES AMSTRAD MAGAZINE

PRÉMIOS

1º Lugar — Aparelhagem AMSTRAD MCD7 com Compact Disc
2º Lugar — Aparelhagem AMSTRAD MS45
3º Lugar — Folha de cálculo: SUPERCALC 3.2 + 1 caixa de disquetes de 5 1/4".

BASES DO CONCURSO:

1) Os programas deverão ser originais podendo ser jogos, programas de gestão ou outros para computadores Amstrad CPC, PCW e PC.

2) Os programas deverão ser enviados em disquete ou cassete acompanhados da respectiva listagem e com as instruções julgadas convenientes para o teste e apreciação do programa.

3) O júri será formado por um grupo de experientes programadores.

4) Os programas enviados e não premiados serão devolvidos. Os programas premiados passarão a ser propriedade do editor para uso que este achar por bem.

5) O prazo de entrega para os programas termina a 30 de Setembro.

6) A acompanhar o programa deve vir BEM LEGÍVEL as seguintes dados:

NOME:
MORADA:
CÓDIGO POSTAL: LOCALIDADE:
TEL.

7) Os programas deverão ser enviados para:
PUBLINFOR, S.A.
Concurso — Amstrad Magazine — Centro de escritórios das Laranjeiras
Praça Nuno Rodrigues dos Santos, Nº 7,
sala 13 2º Piso — 1600 Lisboa

N. B. — Todos os programadores que não obedeçam a estas regras, serão automaticamente eliminados.

POSTAL 6

Simples de compreender depois de uma breve análise do titulo da secção, este espaço dedica-se, tal como aconteceu no número anterior, à explicação de um novo serviço AMSTRAD MAGAZINE. Desta vez vamos falar da assinatura de algumas revistas inglesas.
Como os leitores mais conhecedores da imprensa especializada britânica já sabem há bastante tempo, as revistas dedicadas ao hardware Amstrad, editadas nesse território, são da exclusiva responsabilidade de um único editor. É assim, que tanto a Amstrad Professional Computing, como a Amstrad Computer User e a Amstrad PCW Magazine são colocadas à venda pela mesma distribuidora, são geridas em termos administrativos pelas mesmas pessoas e dispõem de um sistema idêntico de condições de assinatura. Face a tudo isto é fácil fazer a assinatura de qualquer uma das três revistas num unico postal, através de um único pagamento.

Amstrad Professional Computing

Esta revista, sendo uma "publicação-símbolo" das edições **Avralite**, é uma das publicações especializadas britânicas que nenhum utilizador de Computadores Pessoais Amstrad pode dispensar. Ela orgulha-se de possuir, não só uma excelente apresentação gráfica, como também alguns dos melhores jornalistas técnicos da actualidade. A **APC** inclui todas as secções habituais neste tipo de revistas tais como comunicações, educação e todos os aspectos do hardware e software que os leitores esperam encontrar numa revista deste nível.
Para além das secções habituais a **APC** publica ainda regularmente análises detalhadas dos novos produtos lançados no mercado.

A editora

Criada em finais de 1985 para satisfazer os utilizadores de computadores Amstrad, a **Avralite**, manteve-se desde sempre uma editora autónoma, correspondendo desta forma aos ideais que presidiram ao seu processo de formação.

Amstrad Computer User

Actualmente, a **Amstrad Computer User** encontra-se dedicada aos possuidores da sempre popular linha de CPC's. Embora a maior parte da revista assente nos jogos e nas aplicações destinadas à ocupação dos tempos de lazer, as aplicações sérias não são esquecidas, nem as necessidades do programador deste tipo de máquinas, publicando--se regularmente abordagens de temas tão diversos como, por exemplo, a análise de "packages" vocacionadas para o mundo dos negócios, ou a explicação dos mistérios do Assembler.

Amstrad PCW Magazine

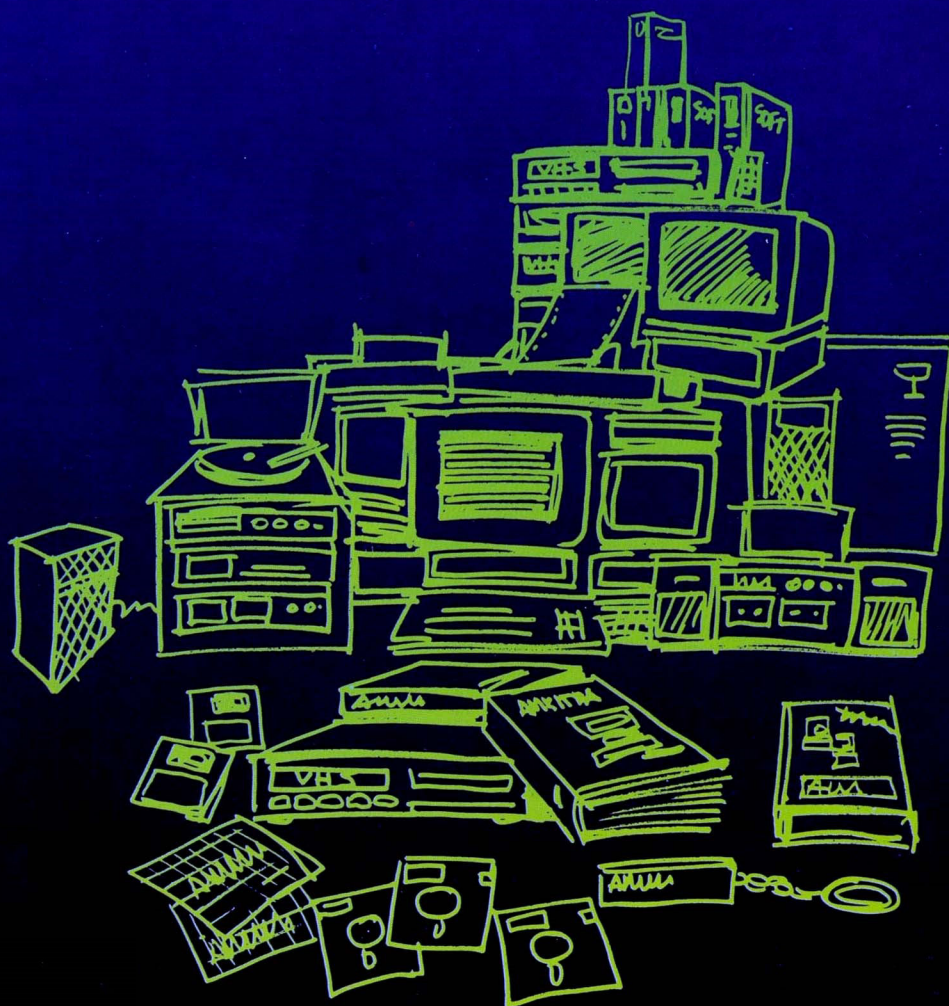
Sendo a mais recente das três revistas em evidência, a **Amstrad PCW Magazine** satisfaz as necessidades de um mercado crescente de utilizadores de máquinas da linha PCW. Ela concentra-se nos usos práticos para este tipo de máquinas, com artigos que permitem aos entusiastas do hardware construirem eles próprios alguns interfaces, e algumas análises de processadores de texto, folhas de cálculo, e "packages" de CAD/CAM.
Quase desde o início do seu lançamento que esta publicação lidera o mercado em termos de revistas para o utilizador do PCW.

A assinatura

Para assinar qualquer uma das revistas referidas (ou qualquer conjunto de revistas de todos os possiveis entre elas) o leitor da AM deve preencher com cuidado o **POSTAL 6** presente nas folhas de cartolina que seguem este artigo, e enviá-lo, dentro de um envelope, para a morada que se encontra no verso do mesmo, acompanhado pelo respectivo cheque para pagamento (em libras esterlinas). Depois, é só esperar alguns dias e lá estará com um carácter regular a sua encomenda na caixa do correio. Tudo muito mais simples do que correr todos os dias até a livraria mais próxima para saber se já chegou o último número da revista X que muitas vezes não conseguimos adquirir porque entretanto já tinham sido vendidos todos os exemplares

clube

AMSTRAD MAGAZINE

REVISTA DOS UTILIZADORES AMSTRAD

FREE SOFT

O Clube dos Leitores da Amstrad Magazine cria neste número uma nova secção — FREE SOFT. As referências incluídas nesta rubrica serão fornecidas aos novos leitores, sem quaisquer encargos para além dos inerentes ao fornecimento das disquetes e aos portes de correio. Cada referência será, pois, fornecida por 350$00 (IVA incluído).
A maior parte dos programas têm a documentação registada na própria disquete, que pode ser impressa ou consultada no ecrã, fazendo TYPE READ.ME > PRN ou simplesmente TYPR READ.ME.
Em muitos casos, o Cube AM traduziu as instruções e assim o leitor poderá aceder--lhes já em português.
Mas, fundamentalmente, o que é o FREE SOFT?
São programas desenvolvidos por particulares que, pelo orgulho de verem os seus trabalhos divulgados (ou a troco de eventuais donativos), autorizam a cópia e distribuição dos seus programas, mas apenas com a simples condição de não serem vendidos.
Em FREE SOFT podem encontrar-se muitos tipos de programas, mas na maior parte dos casos, os títulos com mais divulgação dizem respeito a cálculo, estatística, utilitários e jogos.
O Clube AM abre esta nova secção ainda com um número reduzido de items, mas garante uma lista muito maior para o próximo número.
NOTA: Utilise o POSTAL 4 para os pedidos de FREE SOFT.

JOGOS DIVERSOS

BUGS!

Uma centopeia anda à sua caça pela floresta. Quando você atira e acerta a criatura multiplica-se. É um jogo de rapidez.

Barra de espaços — Prime-se para começar o jogo.
F10 — Atira.
F9 — Atira continuamente.
->, <-, f, ! — Movimentos do cursor.
ESC — Para terminar.

REF. S01

PINBALL

Jogo similar ao "flipper". Tem bons efeitos gráficos.
Podem jogar de 1 a 4 pessoas. Isso é definido no início do jogo primindo-se a tecla da direita tantas vezes quantos os jogadores. Para se iniciar o jogo prime-se a tecla da esquerda e a bola é atirada. Para defender as bolas as teclas são, novamente, da esquerda e da direita. Para terminar prime-se ESC.

REF. S01

PITFALL

REF. S01

Voo movimentado e rápido sobre vales ou abismos. O jogador necessita ter reflexos rápidos para evitar paredes e outros perigos. Para os movimentos primir ->, <-, f, !. Conseguem-se mais pontos primindo !. São permitidos até 10 choques nas paredes sem que o aparelho rebente. Contudo, depois de cada 3 000 pontos, dá-se um bónus de 5 choques. Tocar numa estrela dá um bónus de 200 pontos. Cair numa face é morte instantânea.

POKER MACHINE

Jogo de poker com máquina em que se puxa a alavanca. Há 9 preços diferentes que vão de 1 a 9. O jogo começa com 200 pontos. Por cada jogo deduzem-se ao montante de pontos o custo do jogo.

NOTA: Se tiver uma placa Hercules necessita, antes de jogar, correr o utilitário CGA2. Para que a velocidade dos jogos fique mais lenta, faça correr o utilitário AMSTRAD (isto se o seu computador tiver um processador 8086).

REF. S01

PYRAMID

Jogo de movimentos rápidos onde você salta para cubos para evitar súbitas pedras e criaturas.
O objectivo do jogo é mudar a cor de todos os cubos enquanto se evitam os perigos: Para alterar a cor ao primeiro cubo prima a barra de espaços. Os outros mudam de cor quando se salta para eles. As teclas de movimento são:

U I

Z X

REF. S01

RAIN

Jogo similar ao "flipper" só que em vez de se jogar com uma bola de cada vez, joga-se com uma "chuva" de bolas. Tem bons efeitos gráficos.
Podem jogar de 1 a 4 pessoas. Isto é definido no início do jogo primindo-se a tecla da direita tantas vezes quantos os jogadores. Para se iniciar o jogo prime-se a tecla da esquerda e as bolas são lançadas. Para defender as bolas as teclas são, novamente, da esquerda e da direita.
Para terminar prime-se ESC.

REF. S02

ROCKETS

Fascinante e rápido jogo de batalhas no espaço.
Barra de espaços — Dispara lasers.
Teclas de cursor — Movimentam a nave em 8 direcções.
"C" — Limpa os detritos no espaço.
"S" — Abranda a velocidade.
"F" — Acelera a velocidade.
"P" — Faz parar o jogo.
ESC — Para terminar.

REF. S02

XWING

REF. S02

Uma boa versão dos jogos do tipo Guerra das Estrelas onde você luta contra "Darth Vader" e a estação espacial "Death Star".

NOTA: Se tiver uma placa Hercules necessita, antes de jogar, correr o utilitário CGA2. Para que a velocidade dos jogos fique mais lenta, faça correr o utilitário AMSTRAD (isto se o seu computador tiver um processador 8086).

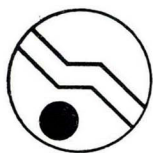
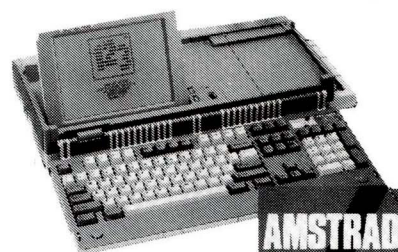
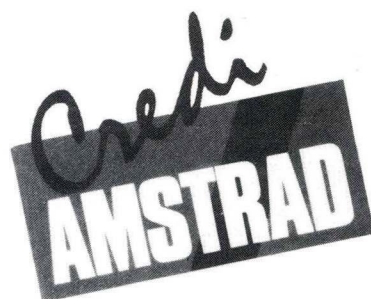
MAHJONG

O jogo de MAHJONG é similar ao jogo de cartas "RUMMY".
Cada jogador vai tomando cartas do baralho (ou ladrilhos como se diz no MAHJONG) e depois vai descartando até obter 4 conjuntos completos e um par. Os conjuntos chamam-se PUNG (3 ladrilhos idênticos: 3 ventos norte, 3 dragões vermelhos, etc.), KONG (4 ladrilhos idênticos: 4 bamboos n. 6, 4 ventos sul, etc.) e CHOW (sequência de 3 ladrilhos do mesmo naipe, por exemplo).
No MAHJONG há 3 naipes: caracteres, circulos e bamboos. Cada naipe está numerado de 1 a 9. Além destes, há ladrihos especiais: os 4 ventos (leste, sul, oeste e norte) e os 3 dragões (branco, verde e vermelho). Joga-se com 4 destes conjuntos o que perfaz 136 ladrilhos.

Como jogar:

Há 4 participantes. O jogador, que é controlado por si, o debaixo, o oposto e o de cima, sendo estes 3 controlados pelo computador.

A mão do jogador situa-se na parte inferior do ecrã e as suas opções (quais as teclas que podem ser pressionadas) estão sempre em linha. O jogador tem 13 ladrilhos que são postos em ordem automaticamente. Quando chegar à sua vez é apresentado um ladrilho à direita do ecrã. Ou fica com ele ou descarta-o.
Para declarar MAHJONG tem de de ter 14 ladrilhos arranjados do seguinte modo: 4 conjuntos de PUNG, KONG ou CHOW e um par único. O último ladrilho pode ser obtido ou porque lhe calha em vez ou pode ser reclamado a qualquer dos outros jogadores.
Para se saber quanto ganhou o vencedor da mão, adicionam-se os pontos obtidos por todos os jogadores perdedores e dobra-se esta base dependendo de certos ladrilhos ou combinações de ladrilhos. Este total é adicionado ao vencedor e deduzido a cada um dos outros jogadores.
O limite imposto para cada mão é de 1 000 pontos para evitar somas muito altas. Quem detiver o vento leste paga ou recebe a dobrar. Um jogo é composto por 4 mãos (excepto para os principiantes que é composto por 2).
Há um limite de tempo para jogar, quem não o fizer nesse intervalo perde a vez.

REF. S02

PROGRAMAS ESTATÍSTICOS

REF. S03

STAR-SAK

É um útil instrumento suplementar para quem analisa dados frequentemente e tem acesso a variadas packages estatísticas. Este programa não faz cálculos que requerem dados observados. Distribuições e testes que se podem utilizar: distribuição normal, distribuição T, distribuição do qui-quadrado, distribuição F, teste de independência/homogeneidade de proporções numa tabela de 2 entradas, teste de Mantel-Haenszel, teste de Mcnemar, coeficientes de correlação e teste de Bartholomew para proporções crescentes.

REF. S03

PC-SIZE

É um programa que determina dimensões de amostras.
PC-SIZE pode calcular a potência do tamanho de amostras específicas ou deteminar o tamanho necessário para se conseguir uma determinada potência.

PC-PITMAN

PC-PITMAN faz testes a números aleatórios e testes U de Wilcoxon-Mann-Whitney na presença de um número arbitrário de ligações entre os dados.

REF. S03

FORGET-IT

FORGET-IT edita graficamente o que foi introduzido por Tukey como uma técnica gráfica para representar a interacção de estruturas numa tabela de duas entradas.

REF. S03

PC-PLAN

PC-PLAN gera planos aleatórios de tal forma que se torna fácil editá-los num processamento de texto para incorporação em memos e relatórios.

REF. S03

PC-MULTI

Constrói intervalos de confiança simultâneos para as diferenças entre pares de médias usando as diferenças significativas de Tukey.

NOTA: São necessárias 2 diskettes. Os programas estão escritos em Fortran.

REF. S03

PC-EMS

Este programa usa o algoritmo de Cornfield e Tukey (1956) para calcular tabelas de quadrados médios esperados para balanços de experiências.

REF. S03

MATH PAK

MATH PAK é um programa que ajuda e acelera algumas das muitas operações matemáticas. Ele é dirigido especialmente para estudantes do ensino secundário, superior e para qualquer pessoa que necessita de ajuda matemática.

Opções disponíveis:

— ADIÇÃO — Adiciona até 60 valores de cada vez.
— SUBTRAÇÃO — Permite a sub. dos valores que se quiser.
— MULTIPLICAÇÃO — Permite a mult. dos valores que cada um quiser.
— DIVISÃO — Divide quaisquer 2 valores.
— ÁREA — Calcula a área de circulos e triângulos.
— FORMULAS QUADRÁTICAS — Cálcula as 2 raízes de fórmulas quadráticas.
— TRIGONOMÉTRICAS — Fornece ao utilizador conversões trigonométricas entre as quais o seno, coseno e tangente para graus e radianos. Também fornece uma tabela trigonométrica (em graus).
— CONVERSÃO HEXA/OCTAL — Converte números décimais em hexadécimais e octais.
— CONVERSÃO BINÁRIA — Converte números binários em décimais.
— CONVERSÕES MÉTRICAS — Converte cm em polegadas, metros em pés, graus Fahrenheit em Celsius, km em milhas, litros em galões, gramas em onças, litros em kg. E vice-versa.
— JOGOS — Fornece um jogo de base matemática que desafia o utilizador na procura de respostas. Há 3 tentativas e 5 níveis de dificuldade.
— DETERMINANTES —
— Permite saber se 2 linhas se interceptam num determinado intervalo.
— Dando 2 pontos encontra a equação; usando a equação ay+b=ms+d edita os pontos.
— Resolve 2 equações a 2 incógnitas.
— Resolve 3 equações a 3 incógnitas.
— FÓRMULAS E CONVERSÕES — Edita as fórmulas para cálculo das áreas do quadrado, rectângulo, cubo, trapézio, paralelogramo, etc. Tem uma tabela de conversões (jardas em metros por ex.).
— CONVERSÃO DE COORDENADAS POLARES EM RECTANGULARES — Introduzem-se 2 valores representativos do ponto A (r, @). O valor r representa a distância da origem ao ponto A e @ o ângulo (em graus) do ponto. O computador dá o X e Y.
— CONVERSÃO DE COORDENADAS RECTANGULARES EM POLARES — É o inverso da opção anterior.
— PONTOS F (x) — Encontra valores de f(x) dos valores de x de -5 a 5. A fórmula utilizada é f(x)=ax^5+bx^4+cx^3+dx^2+ex+k. Também lista os valores da primeira e segunda derivada da função.
— LOGARÍTMOS — Cálcula logarítmos de base 10 ou de uma base "a". Sendo f(x)=a(bx^c+dx^e+k)^n.

NOTA: O programa está escrito em Basic. O símbolo "^" significa elevado a.

REF. S04

EPISTAT

Este programa dirige-se a professores e alunos de estatística, a quem se interessa pela estatística ou pela informação que esta dá no conhecimento de certas realidades da empresa e na tomada de decisões.
EPISTAT ajuda o utilizador a escolher o melhor teste para os seus dados e faz correr um conjunto de testes e análises.
Para a maioria dos testes é necessário introduzir as amostras e/ou variáveis na opção "DATA-ONE". Introduz-se o nome da variável (ou amostra) e os dados observados. O computador cálcula e edita de imediato a média, a mediana e o desvio padrão. Estes dados são salvaguardados num ficheiro e utilizados posteriormente em testes, em análises, em gráficos.

As opções disponíveis são:

— TEOREMA DE BAYES — probabilidades dos testes darem falsos positivos e falsos negativos; probalidade de uma doença dar um teste-positivo.
— DISTRIBUIÇÃO BINOMINAL — Tabela a uma aba.
— TESTE DO QUI-QUADRADO — Avalia uma tabela de dados ou um valor do qui-quadrado conhecido.
— COEFICIENTES DE CORRELAÇÃO — De Pearson e de Spearman.
— TESTE DE FISHER — Avalia tabelas de 2x2 de variáveis discretas.
— HISTÓGRAMAS — É feito com dados introduzidos.
— REGRESSO LINEAR — Cálcula a linha de regressão dos mínimos quadrados para pares de amostras; faz transformação de dados através de uma função.
— TESTE DE MANTEL-HAENZEL DO QUIQUADRADO — Avalia a relação entre 2 variáveis discretas enquanto controladas pelo efeito de uma terceira.
— TESTE DE MANTEL-HAENZEL DO QUI-QUADRADO PARA MÚLTIPLOS CONTROLES — Compara, avaliando, uma amostra com 2 ou mais amostras de controle.
— TESTE DE MCNEMAR — Avalia tabelas de 2x2 de pares de variáveis discretas utilizando a correcção de Yates.
— DISTRIBUIÇÃO NORMAL — Tabela a 2 abas; compara a média da amostra com a média da população.
— DISTRIBUIÇÃO DE POISSON — Tabela.
— TESTES DE CATEGORIAS não paramétricos) — Teste de Wilcoxon, para amostras independentes; teste do sinal, para amostras emparelhadas.
— SCATTERGRAM — Traça a linha de regressão entre 2 variáveis.
— TESTE T — Compara as médias de 2 amostras.

Além disto, EPISTAT, permite a transferência de dados entre 2 dos seus ficheiros e de dados introduzidos num ficheiro Fortran (ou SDF) para um ficheiro EPISTAT. Outra das suas potencialidades é ter um gerador de amostras com números aleatórios que ajuda na selecção de amostras aleatórias para diferentes finalidades. Outra ainda é ter uma opção que cálcula o tamanho necessário aproximado para se atingir um determinado nível de significância estatístico dado.

REF. S04

MAHJONG — Para ecrã EGA

O objectivo do jogo é retirar tantas peças do tabuleiro quantas forem possíveis dentro das regras do jogo. As peças são retiradas, aos pares, dos cantos esquerdo e direito. Qualquer peça que não se situe nos cantos é considerada bloqueada e não pode ser retirada. As peças estão colocadas em cinco níveis tendo cada um canto e assim sendo há mais peças desbloqueadas do que parece à primeira vista.
Há vários conjuntos de peças: caracteres, pontos, bamboos, dragões, ventos, flores e estações. Nos cinco primeiros conjuntos os pares têm que ser constituídos por duas peças exactamente iguais (dois bamboos número um). Nos dois últimos, desde que as peças pertençam ao mesmo conjunto, os pares podem ser feitos (o Outono calha com o Verão). Há quatro séries iguais dos cinco primeiros conjuntos e apenas uma dos dois últimos.

Como jogar:
O tabuleiro é uma grelha numerada. Para se retirarem os pares tem de se introduzir a coluna e a linha onde as peças estão colocadas. Se depois de se seleccionar uma peça se mudar de ideias basta premir "return" e introduzir as novas coordenadas.
Outros comandos disponíveis:
Q: Sai do jogo
H: Ajuda — Na primeira peça dá as combinações existentes
Na segunda dá o par para a primeira peça (caso exista)
S: Salvaguarda o jogo
R: Faz o restauro do jogo
N: Começa um novo jogo
A: Repõe o mesmo jogo (mesma arrumação de peças)
Linhas de comando:
— Pode escolher-se um determinado tabuleiro introduzindo MAHJONGG — bx (onde x vai de 0 a 65535).
— Introduzindo MAHJONGG — tx (onde x é o número de minutos para cada jogada) pode jogar-se com um contador de tempo para fazer competições.
Quando se quer competir com alguém e para que os dados fiquem guardados é necessário seleccionar T no primeiro ecrã. Nesta opção são introduzidos os nomes dos jogadores.

Nota: É possível jogar com o rato.

REF. S04

ALLMAC

Este programa permite que se façam desenhos constituídos por um conjunto de pontos (576 em largura e 720 em altura).
ALLMAC funciona em diferentes ecrãs:
— Controlador CGA com monitor de média resolução;
— Controlador EGA com monitor de média resolução;
— Controlador EGA com monitor EGD;
— Controlador e monitor monocromáticos (não é possível visualizar o desenho mas pode-se editá-lo e imprimi-lo).
E em diferentes impressoras:
— EPSON;
— Panasonic/Super 5,
— Apple DMP/NEC 8023,
— Gemini 10,
— IBM Graphics Printer,
— C. Itoh 8510,
— Toshiba P321
— HP Laser jet.
No ecrã principal podem ver-se as opções disponíveis e o que fazem, a "fotografia" do desenho editado no momento em tamanho reduzido e o seu nome.
As opções disponíveis são:
D — Lista o que está no directório
L — Carrega para a memória um desenho
P — Imprime o desenho editado
S — Reconfigura a impressora
V — Visualiza o desenho em tamanho normal
W — Salvaguarda o desenho no disco
E — Muda o programa para o modo de edição. É nesta opção que se fazem os desenhos.
É muito simples desenhar pois para isso basta utilizar as teclas de movimento e a barra de espaços. Quando se quer um ponto preto prime-se a barra de espaços e de seguida a tecla de movimento. Quando se quer um ponto branco basta premir a tecla de movimento.

REF. S04

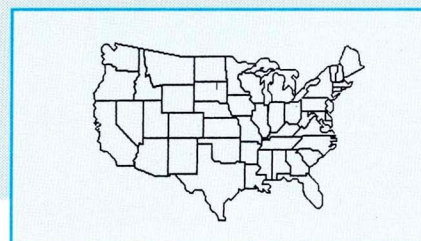
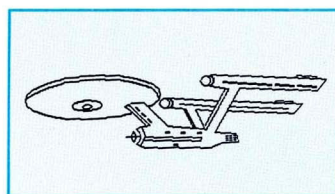
ICON MAKER

Este programa é útil para quem desejar fazer desenhos ou letras para depois os introduzir na área de dados de um programa em Assembler. Podem escolher-se três níveis de resolução gráfica aos quais está associado o número de cores disponíveis. Preto e branco, alta resolução; 4 cores, média resolução e 16 cores, baixa resolução. Cada desenho pode incluir até 100x120 pontos.

REF. S04

ALTAMIRA — Editor gráfico

REF. S05

Este programa permite, movimentando o cursor, desenhar no ecrã do computador, linhas, circulos, etc. Depois de desenhado o objecto pode-se mudar a sua cor, o seu tamanho, rodá-lo, movimentá-lo, copiá-lo para outro sítio do ecrã ou ainda apagá-lo. Todo o controlo é feito através do teclado. As teclas de movimentação fazem o cursor mover-se de um lado para o outro e os comandos de uma letra desenham os objectos. Por exemplo, prime-se a letra 'e' quando se quer desenhar uma elipse, a letra 'o' quando se quer desenhar um círculo, etc.
Podem-se desenhar caixas, caixas preechidas, linhas, linhas desenhadas com pontos, linhas desenhadas com traços, vectores, pontos, círculos, círculos desenhados com traços, círculos preenchidos, arcos, polígonos, tracejado vertical e horizontal, elipses, elipses preenchidas e texto. Também se pode alterar a cor do fundo do ecrã, à distância que o cursor percorre em cada movimento e manipular grupos de objectos.
Podem adicionar-se dois desenhos e também visualizar um enquanto se edita outro.
Os desenhos podem ser salvaguardados em ficheiros como "mapas bit" ou, em alternativa, ser traduzidos para Pascal podendo assim ser incorporados em programas (a versão do Turbo Pascal tem de ser a 2.00 ou superior).

Nota: Este programa corre em ecrãs mono e policromáticos. Necessita 128k ou 192k de memória perdendo no primeiro caso algumas potencialidades.

DRAW POKER

Conhece as regras do poker de cartas e gosta de jogar? Então este é um jogo indicado para si.

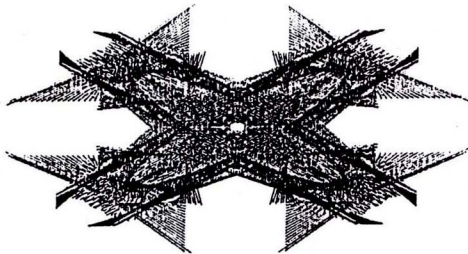
Algumas regras:
— Por cada jogo o jogador paga 2 dólares
— Cada aposta custa pelo menos 20 dólares
— Um baralho de 52 cartas, não há Joker
— A cada jogador são dadas 5 cartas que podem ser trocadas no decorrer do jogo
— Se um jogador é apanhado a fazer bluff perde
Para que o jogo comece digite "DRAW" e faça "ENTER". É-lhe então pedido o seu nome (no máximo até 14 caracteres). DRAW vai salvaguardar o seu lucro ou perda num ficheiro particular que procura no drive. Quando o jogo começa, ficará contra 6 oponentes (do computador) que lhe tentarão ficar com o seu dinheiro o melhor que puderem. A si vão-lhe sendo pedidos os dados para cada jogada (se quer apostar, se quer abrir o jogo, se quer trocar as cartas e quais, etc.).
A partir do momento que decidir jogar tem que o fazer pelo menos uma vez.
Primindo a tecla errada enquanto joga uma mão a regra é a seguinte: "Uma vez que falou, está falado".
DRAW POKER permite fazer hardcopy dos jogos passados. Ao mesmo tempo que cria um ficheiro para cada jogador também cria um ficheiro onde guarda o melhor jogo de cada jogador. A hardcopy mostra o número do jogo, a data em que se jogou, o número de mãos nessa sessão, os ganhos ou perdas, o total dessa data e finalmente a média do jogador.

REF. S05

PIANO MAN

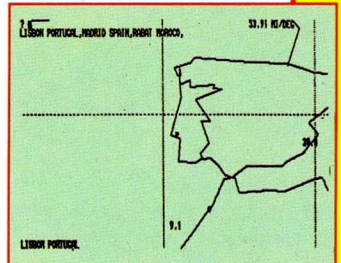
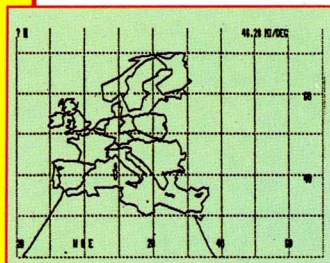
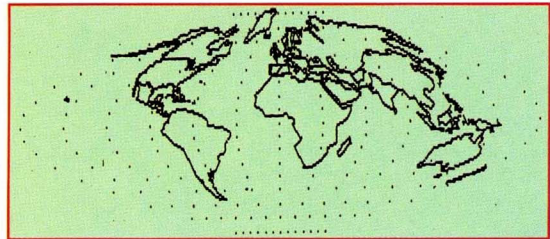
PIANO MAN é um programa espectacular para os apreciadores de música e principalmente para quem gosta de compor. O Trabalho dos compositores fica muito facilitado na medida em que lhes basta tocar a música e esta fica guardada em disco com um nome escolhido.
PIANO MAN permite que se utilize o teclado como um piano. Pode tocar-se uma música simplesmente ou gravá-la ao mesmo tempo. O utilizador pode tocar em legato, onde o som de uma nota só acaba quando a outra é tocada, ou como um piano normal.
Toda a música que se grava pode ser alterada, quer nota a nota, quer em bloco.
Estando a música visualizada no ecrã, é possível alterar qualquer nota (por exemplo, substituir-se um dó# por um ré), a oitava em que essa nota é tocada (estão disponíveis 7 oitavas), a duração da nota (o tempo é medido em milésimos de segundo) e por fim o valor do staccato (se 2, por exemplo, 2/10 é a duração do silêncio). Além disto é possível acrescentar notas à música, apagar notas e acrescentar silêncios.
As alterações permitidas em bloco são similares às de nota a nota. São elas: alterar a(s) oitava(s) em que se toca toda a música, o staccato e a duração.
Por vezes hesita-se muito tempo para tocar uma nota; PIANO MAN tem uma opção que permite dar a todas as notas um tempo de duração razoável (round off). Pode ser útil correr esta opção antes de tocar a música para ver como ficou antes de a editar para alterações.
Para músicas muito longas aconselha-se a gravação em secções, podendo estas ser combinadas através do chamamento para memória das diferentes partes e salvaguardado o conjunto num só ficheiro. À medida que se vai compondo pode ir-se ouvindo a música. O número limite de notas é de 3 700 para quem tiver 138k de memória (no Amstrad 1512 dá para mais de 25 000).

UTILITÁRIOS PARA ECRÃS EGA

EGAINFO — É um documento que dá informações sobre os ecrãs. EGA e os programas que correm nele ou não.
EGATECH — Documento com as especificações técnicas dos ecrãs EGA.
ST12 — Utilitário que edita ficheiros no ecrã em modo "scroll" permitindo ao utilizador regular a sua velocidade: mais devagar, mais depressa, parar, tornar a andar.
ROLEX — Relógio com números, ponteiros, calendário; um autêntico relógio Rolex. Com ele se vê a data e hora do sistema e se avalia a resolução gráfica dos ecrãs EGA.
EGAKAL — Caleidoscópio colorido. Pode ficar-se horas entretido a olhar para o fluir dos desenhos e cores. O desenho que se segue apenas dá, a nível de forma, uma ideia muito reduzida do que é este caleidoscópio.

8X6 — Quando executado, o ficheiro fará com que um ecrã equipado com um sistema EGA edite 80 colunas e 60 linhas em vez das usuais 80X25. Alguns programas trabalham bem com isto, outros simplesmente ignoram-no. De qualquer forma é um pequeno utilitário que permite visualizar no ecrã mais informação do que normalmente é possível.
EGAART — Mais um utilitário para se avaliarem as capacidades de resolução gráfica de um ecrã EGA. Constroi um conjunto de desenhos.
EGAITAL — Põe o texto do ecrã em itálico.
EGAMODE — Altera o modo de um ecrã EGA. Modo 0 (letras grandes), modo 3 (letras pequenas).

REF. S05

WORLD

Este é um interessante programa de geografia. Desenha o globo terrestre e os seus países em variadas escalas.
Há muitas formas de se pedir o traçado de um determinado país ou área. Pela longitude e latitude; países da Europa; Estados Unidos da América; cidades capitais dos Estados Unidos; outras grandes cidades dos Estados Unidos; grandes zonas como Rússia, Canadá, África, Austrália, etc. e capitais de países, estando associadas 166 capitais aos respectivos países. WORLD para além de desenhar o país ou zona também edita a latitude e longitude do centro do ecrã e o número de milhas num grau da latitude do centro do ecrã.
O globo terrestre pode ser visualizado de vários ângulos.
Ainda outra opção disponível para se conseguir ver determinada zona do mundo é editar o mapa-mundi, posicionar-se no ponto que interessa e aumentar essa área (efeito zoom).
Quando se editam os mapas é possível saber os furacões que ocorreram nessa zona e os seus rastos e ainda introduzir os furacões que se deram numa determinada latitude e longitude numa dada data.

REF. S05

MUSIC

Este programa é constituído por um conjunto de 72 músicas.

— 9 to 5
— All Notes
— Random Blues
— Centerfold
— 2001: A Space Oddessy
— Ever Onward
— The Master Warbler
— Raindrops Keep Falling On My Head
— Solfeggietto
— Symphony #5
— Symphony #40
— Every Breath You Take
— Popcorn
— Airplane
— Rail
— William Tell Overture
— Yesterday
— Mexican Hat Dance
— Teh Sleeping Beauty Waltz
— Romeo and Juliet
— Toccata and Fugue in D Minor
— Prelude in C Major
— Baby Elephant Walk
— Moon River
— El Choclo Tango
— Waltz of the Flowers
— Clavier
— Anitra's Dance
— Sonata IV in E Major, 1st Mov.
— Minuet in D Minor
— Minuet
— Dixie
— Ungeduld
— The Irish Ballad
— Sailor's Hornpipe
— I am the Walrus
— Garota de Ipanema
— Cabaret
— Downton
— The Impossible Dream
— Satisfaction
— Sounds of Silence
— Sunrise, Sunset
— Moonlight Sonata, 2nd Mov.
— Moonlight Sonata, 3rd Mov.
— Take Five
— Sweet Georgia Brown
— Gizmo's Song from Gremlins
— The Addams Family
— The Odd Couple
— Speak Softly Love
— Star Wars
— Red Roses for a Blue Lady
— Funeral March
— Movin' Right Along
— The Little Notebook for Bach
— The Birth of the Blues
— Glow Worm
— British Grenadiers
— O Canada
— One More River
— Cockles and Mussels
— Old Folks at Home
— Trumpet Song
— Silent Night
— Deck the Halls
— The Turtle Song
— Jingle Bells
— On Come, All Ye Faithful
— Making Our Dreams Come True
— Miniac
— MultiSound

Nota: O utilizador necessita ter o Gwbasic.

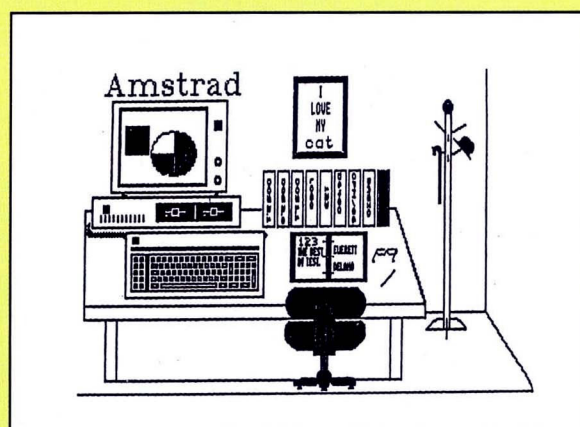
REF. S06

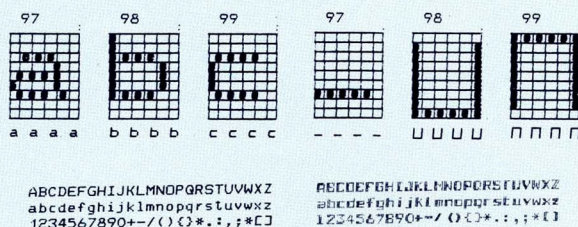
PAINT

REF. S06

Este programa é um bom auxiliar para quem tenha de fazer uma apresentação de um determinado conjunto de ecrãs desenhados.

PAINT tem três módulos.
No primeiro definem-se quadrados, círculos, semi-rectas. Muda-se a cor a blocos. Definem-se "moldes" dos desenhos já existentes, que pode ser guardados e escritos quer no próprio desenho quer noutros.
tudo o que se desenha é possível apagar.
No segundo módulo é como escrever com uma caneta. Depois de se definir a cor com que se quer desenhar fazem-se os traços desejados.
No terceiro escrevem-se letras e números. Há cinco tipos diferentes de letras. Os números podem-se escrever da esquerda para a direita, de baixo para cima e de cima para baixo.
Depois de feito o desenho ele é salvaguardado. É necessário ter atenção aos nomes que se dão aos desenhos pois na apresentação vão sair por ordem alfabética. A apresentação pode ser manual ou automática. Se for manual pode andar-se com os desenhos de frente para trás e de trás para a frente. Se for automática os desenhos permanecem no ecrã 60 segundos no máximo e a apresentação é sempre seguida. Para que esta funcione é necessário ter o Basica.

Segue-se um exemplo de um desenho feito no PAINT.

FXMATRIX

Com este programa o utilizador pode criar o seu próprio alfabeto ou então alterar algumas letras ou símbolos que lhe não sejam necessários por outros mais úteis.
Cada alfabeto, ou conjunto de caracteres, é definido num ficheiro. Esse ficheiro é carregado quando o utilizador o desejar utilizar e fica sempre em acção até outro ser carregado, isto mesmo que se desligue o computador.
Os ficheiros são um conjunto de caracteres que são definidos numa tabela de duas entradas com 9x6. O utilizador digita os pontos que deseja para formar um caracter.
Para que se torne mais claro o que quero dizer vou apresentar algumas letras do alfabeto normal e a sua substituição por caracteres diferentes e também dois alfabetos:

Nota: Apenas ao nível da impressão há alteração de caracteres. Este programa é útil para quem tem impressoras Epson Fx e JX.

REF. S06

TurboCAD

De instalação fácil, e utilização simplificada como consequência do funcionamento baseado em menus tipo "pop-up" o TurboCAD pode ser o utilitário que você procura para "dar asas á sua imaginação" no dominio do desenho técnico.
Acompanhado por um completo manual que lhe permite entrar sem grandes dificuldades no mundo do Desenho Assistido por Computador, o TurboCAD assegura a compatibilidade com o AutoCAD (uma das "packages" de CAD mais populares entre os utilizadores de computadores), sendo cerca de 9 ou 10 vezes mais económica do que esta última.

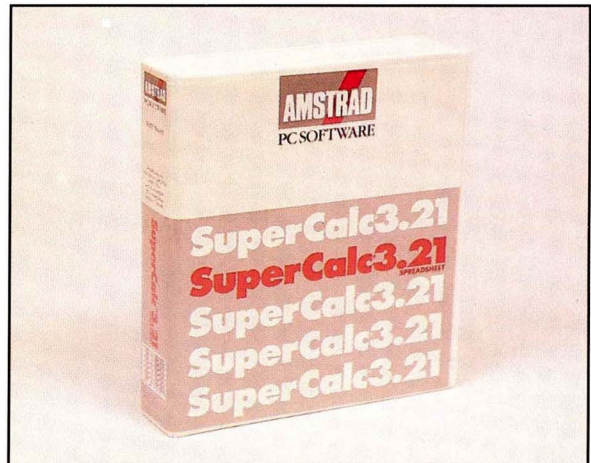
PREÇO: 27 500$00 REF.318, postal 3

Coberturas para impressora AMSTRAD DMP 3000 e DMP 3160

PREÇO:			
	Elicalfe	2 000$00	REF. 202
	Tela plástica	1 755$00	REF. 204
			postal 3

Coberturas para computador AMSTRAD PC 1512 e PC 1640

PREÇO:			
	Elicalfe	4 530$00	REF. 201
	Tela plástica	4 212$00	REF. 203
			postal 3

SUPERCALC 3.2

O standard em folhas de cálculo é, ainda hoje, nitidamente imposto pelo LOTUS 1,2,3. Ninguém sequer coloca isso em causa. O que começa a colocar-se em causa são as vantagens de utilização desta folha de cálculo numa altura em que existem dezenas de outros utilitários com o mesmo fim, compativeis com o LOTUS, mas... muito mais possantes.
É este, por exemplo, o caso do SuperCalc, agora disponivel na sua versão 3.2.
O SuperCalc foi uma das "packages" que soube tirar proveito do facto de não "rasgar" mercado.
Aproveitando os resultados das experiências dos seus "adversários", o SuperCalc 3.2 melhorou muitas das suas caracteristicas, apresentando por exemplo, entre muitas outras qualidades dignas de nota, modos de representação gráfica superiores aos que a maior parte dos utilitários deste tipo incluem, uma boa velocidade de processamento de dados, e um conjunto de "HELP screens" mais do que suficiente para se começar a tirar proveito da "package", mesmo antes de se começar a ler o detalhado manual que a acompanha.
Conclusão: se nunca utilizou uma folha de cálculo, o SuperCalc 3.2 é-lhe indispensável; se já utiliza uma folha de cálculo o SuperCalc 3.2 é-lhe indispensável.

PREÇO: 19 900$00 REF.319, postal 3

CLUBE DOS LEITORES

TODOS OS OS PREÇOS INCLUEM O TRANSPORTE E O I.V.A. A 17%

Porta-chaves AMSTRAD com inscrição das letras a vermelho

PREÇO: 140$00 REF.100, postal 3

DISKETTES AMSTRAD

Em 3", 3.5", ou 5.25" as diskettes Amstrad são fornecidas em conjuntos de 10 unidades com caixa plástica, garantindo uma perfeita formatação e fiabilidade dos dados armazenados.

3"	PREÇO: 8 490$00	REF. 315	postal 3
3.5"	PREÇO: 5 990$00	REF. 316	postal 3
5.25"	PREÇO: 2 690$00	REF. 317	postal 3

EXCLUSIVO DO CLUBE DE LEITORES

JÁ NÃO PRECISA DE SAÍR DE CASA PARA IR JOGAR POKER AO CASINO

O jogo Good Luck é uma réplica do popular Poker das máquinas dos casinos, permitindo todo o tipo de jogadas — 2 pares, sequência, fullen, etc. e, para os mais destemidos, dobrar ou perder

PREÇO: 2 000$00 REF.306, postal 3

GRAVADOR DE VIDEO AMSTRAD

Quantas vezes já desejou gravar aquele filme que passou na televisão e não o pode fazer por não possuir um gravador de video?

Quantas vezes já desejou ver aquele filme que em tempos não viu no cinema e não o pode fazer por não possuir um gravador de video?

Quantas vezes já desejou não ver aquilo que viu na televisão e não teve alternativa por não possuir um gravador de video?

Quantas vezes já desejou ter um gravador de video porque não tem um gravador de video?

Um gravador de video é, de facto, quase indispensável para quem tem um gravador de video, e uma falta imperdoável para quem não tem um gravador de video. É claro que quando não se tem um gravador de video e se pensa em adquirir um gravador de video sem o comparar com o gravador de video que lhe propomos, então mais vale nem pensar em adquirir um gravador de video, porque pesando a relação preço/qualidade O GRAVADOR DE VIDEO AMSTRAD é simplesmente um excelente...

PREÇO :84 900$00 REF.402, postal 3

NOVIDADE

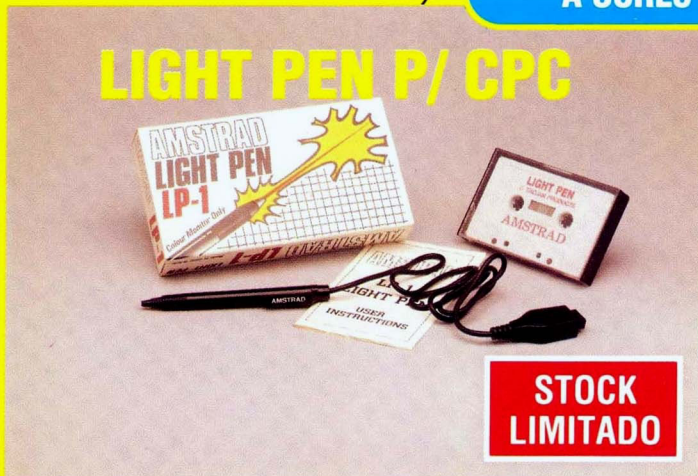
MANUAL DO PC EM PORTUGUÊS

Será que os computadores só podem ser utilizados por quem sabe inglês?
É evidente que não. Embora o conhecimento da língua inglesa facilite a aprendizagem, nunca se poderá considerar indispensável para este efeito. No nosso país, são cada vez mais frequentes as marcas que traduzem os manuais e as packages, e adaptam os teclados, para poderem possuir boas soluções informáticas em mercados que nada têm a ver com a língua inglesa.
Foi assim, seguindo esse princípio, que AM optou por incluir nesta secção a tradução do MANUAL DO PC, para facilitar a vida a todos os que em Portugal preferem ler em português.

PREÇO: 1 900$00 REF. 310, postal 3

SÓ PARA MONITOR A CORES

LIGHT PEN P/ CPC

STOCK LIMITADO

A imaginação não tem limites, mas a forma de aplicar essa imaginação por vezes é limitada por falta de meios adequados.
A LIGHT PEN, uma vez ligada ao CPC, permitir-lhe-á demonstrar as suas capacidades como desenhador, ou caricaturista, por exemplo, possibilitando-lhe a criação de desenhos no écran sem instruções complicadas.
Pegue na caneta e desenhe, ou escreva, no écran aquilo que lhe apetecer. Se por um acaso se enganar, apague e rectifique o trabalho as vezes que desejar.
Não limite a sua imaginação.

PREÇO: 1 999$00 REF. 405, postal 3

CM1 — CONJUNTO DE 5 JOGOS SORTIDOS PARA CPC

?

Se é possuidor de um CPC, se tem entre 5 e 95 anos, se tem tempo para jogar e não tem jogos — então tem um grave problema.
Felizmente nós propomos-lhe uma solução.
5 Cassetes com 5 jogos (surpresa) diferentes, vão diverti-lo por muito mais de 5 horas e custar muito menos de 5 contos, embora também custem um pouco mais de 5 escudos.

PREÇO: 990$00 REF.313, postal 3

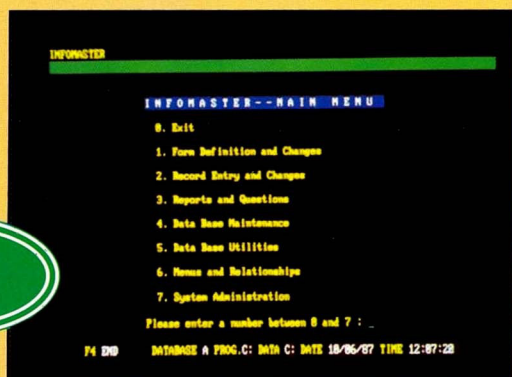
FORTH P/ CPC

Num momento em que começam a surgir no mercado alguns processadores que possuem como linguagem "natural" o FORTH, torna-se interessante poder oferecer aos possuidores dos CPC a hipótese de experimentar o poder desta linguagem como forma de comunicar com a máquina. Com algumas vantagens sobre o BASIC (nomeadamente uma maior velocidade de processamento), o FORTH continua a manter inúmeros adeptos entre os programadores e utilizadores de computadores, que não hesitam em defendê-lo, em muitas situações, como uma das melhores linguagens de programação.

APRESENTADO EM CASSETE

PREÇO: 900$00 REF.314, postal 3

INFOMASTER

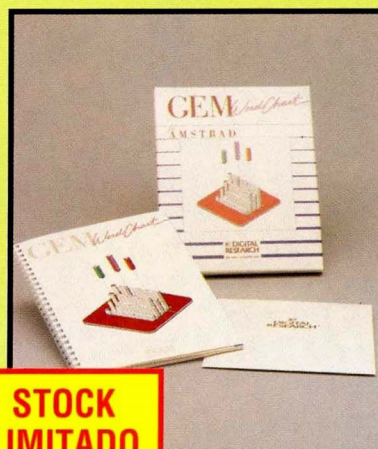
PC

Em bases de dados, é verdade que o DBASE criou um standard, mas não é menos verdade que INFOMASTER ultrapassou esse standard.

Permitindo uma utilização eficiente após alguns minutos de trabalho, possibilitando a utilização de um máximo de 65535 registos em cada ficheiro, e um máximo de 255 campos em cada retgisto, o INFOMASTER torna-se o sistema de gestão de base de dados mais adequado para as pequenas empresas.

Funcionando num sistema de menus que permite a fácil manipulação de informação, e a configuração da base de dados por utilizadores com um mínimo de conhecimentos, esta package utiliza parte da RAM como cache, conseguindo deste modo uma velocidade que em determinadas situações se pode considerar cerca de 400% superior à das bases de dados convencionais.

PREÇO: 4 900$00 REF. 309, postal 3

PC

GEM WORDCHART

STOCK LIMITADO

Actualmente, mais de 80% das apresentações são feitas através de palavras — e não de gráficos.

O GEM WORDCHART, concebido com a intenção de lhe servir de instrumento de trabalho na realização simples de apresentações, permite a utilização de diversos tipos de letras com recurso a inúmeras variantes de cada tipo, selecção de limitadores e formatos, e combinação de cores, através de menus do tipo "drop-down".

Para lhe tornar a composição da folha mais fácil, o texto aparece no écran exactamente igual à posterior cópia impressa, e a largura das colunas pode seleccionar-se com a simples pressão de um botão do "mouse".

Em resumo, o GEM WORDCHART, situa-se entre o PRINT MASTER e o PAGE MAKER, apresentando no entanto, em relação a um e a outro, algumas vantagens na concretização de pequenos trabalhos.

PREÇO: 9 900$00 REF. 308, postal 3

GEM DIARY

PC

STOCK LIMITADO

Aconselhado para os utilizadores do GEM em "regime intensivo", esta package inclui basicamente um calendário perpétuo com agenda, possibilita a utilização de 11 memórias diárias com o display de mensagens em 11 momentos previamente especificados, contendo ainda um pequeno ficheiro, e um diário-bloco de notas. Quase indispensável para quem se move diariamente sobre o GEM, o uso do GEM DIARY torna-se rapidamente um "must" neste ambiente de trabalho.

PREÇO: 4 900$00 REF. 309, postal 3

MCD7 PORTATIL

A fidelidade do compact disk aliada a comodidade dos tradicionais equipamentos de audio encontram-se sob uma única designação: MCD7.

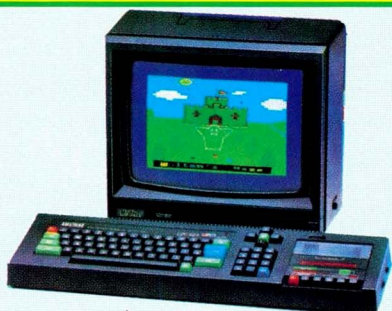
Um amplificador, um sintonizador, um leitor de compacto, um leitor/gravador de cassetes com duas gavetas, e duas colunas laterais amovíveis, é tudo aquilo de que necessita para "beber" calmamente a sua música preferida, enquanto "escuta" o seu whisky com gelo, ou montar uma discoteca à beira-mar, agora que decidiu começar o seu "estágio anual de praia".

PREÇO: 39 900$00 REF. 401, postal 3

DDI-1

Trabalhar num computador que utiliza a cassete como suporte de massa pode tornar-se aborrecido, especialmente depois de se ter trabalhado durante algum tempo com uma máquina que recorre às diskettes para armazenar informação.
A alternativa, numa situação deste tipo, passa quase sempre pela compra de uma nova máquina, ou pela aquisição de uma drive externa. O DDI-1 representa esta última solução para todos os possuidores do CPC 464.
Sendo uma drive de diskettes de 3" com 180Kb formatados, esta unidade torna possível a utilização do CP/M e do LOGO a todos os utilizadores que, por exemplo, já se cansaram do BASIC e do assembler Z 80 como linguagens de programação.

PREÇO: 28 900$00 REF. 404, postal 3

CPC 464
POLICROMÁTICO

Com 64 KB de RAM, 32 KB de ROM, som estereofónico, teclado profissional de 74 teclas, e monitor policromático, o CPC 464 continua a ser uma boa aquisição no mercado dos micros, visto assegurar uma razoável compatibilidade com os micros Amstrad mais recentes, e sofisticados.

PREÇO: 89 900$00 REF. 403, postal 3

REFLEX

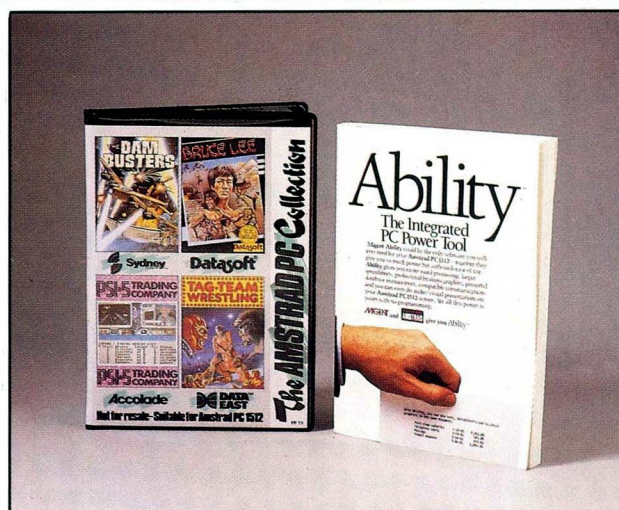
STOCK LIMITADO

PC

Sistema de gestão de base de dados, agenda-plano, package de gráficos, e gestão de listagens e correio, são algumas das muitas capacidades do REFLEX, complementadas pela possibilidade de importar dados directamente das mais conhecidas folhas de cálculo e de outras bases de dados standard.
O REFLEX é sem dúvida o reflexo da aplicação das novas técnicas de concepção de packages, mantendo os necessários elos de ligação com as packages que ao longo do tempo se tornaram standard.

PREÇO: 9 900$00 REF. 307, postal 3

ABILITY + 4 JOGOS

PC

Package integrado de programas que lhe oferece:

a) Base de Dados.
b) Folha de Cálculo.
c) Gráficos de Gestão.
d) Processamento de Texto.
e) Comunicações.
f) Gerador de Apresentações.

Incluindo:

1) Manual de fácil leitura e manuseamento.
2) Utilização compartilhada de dados para as diferentes aplicações.
3) Integração activa entre os programas, (não realizável em programas conhecidos do mercado).
4) Com o programa APRESENTAÇÃO, incluído no Ability, podem preparar-se informações obtidas com os dados manuseados com o programa base.

E ainda 4 Jogos: "The Dam Busters", "Bruce Lee", "Psi 5 Trading Company" e "Tag Team Wrestling".

PREÇO: 8 900$00 REF.301, postal 3

GEM GRAPH + GEM DRAW

GEM GRAPH — Com a simples movimentação do rato e premindo apenas um botão, podemos obter gráficos profissionais de alta qualidade: de barras, tipo tarte com ou sem explosão, de símbolos, de linhas ou de mapas. Do tamanho e estilo que você decidir; com texto, cores e fundos de relevo para dar ao seu gráfico um aspecto tridimensional.
Gem Graph é um programa com excelentes qualidades gráficas.

GEM DRAW — Desenhos lineares, artísticos, organigramas, esquemas, etc. Escolha os elementos no menú e dê largas à sua imaginação. GEM DRAW converterá o seu PC num estúdio profissional com 6 tamanhos e tipos de letra, 20 livrarias de gráficos disponíveis, 39 funções de trama, régua, alinhamento, etc. e quando o seu desenho estiver perfeito, obtenha a cópia impressa em papel ou transparência.

PREÇO: 24 900$00 **REF. 302, postal 3**

SIDEKICK

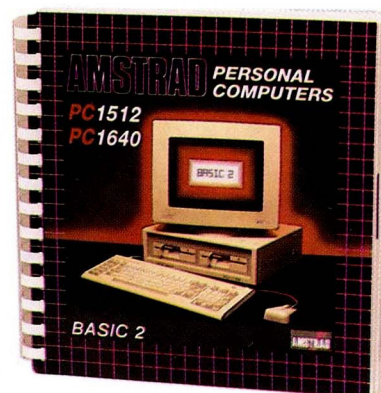
Software concebido para estar instalado no seu computador, em cima da sua secretária para:

- ★ Cálculos rápidos
- ★ Bloco-notas
- ★ Editor de textos compatível WordStar/Turbo Pascal
- ★ Agenda telefónica
- ★ Planeamento de actividades
- ★ Ligação automática de chamadas telefónicas
- ★ Registo de recados e mensagens
- ★ Pesquisa de códigos ASCII

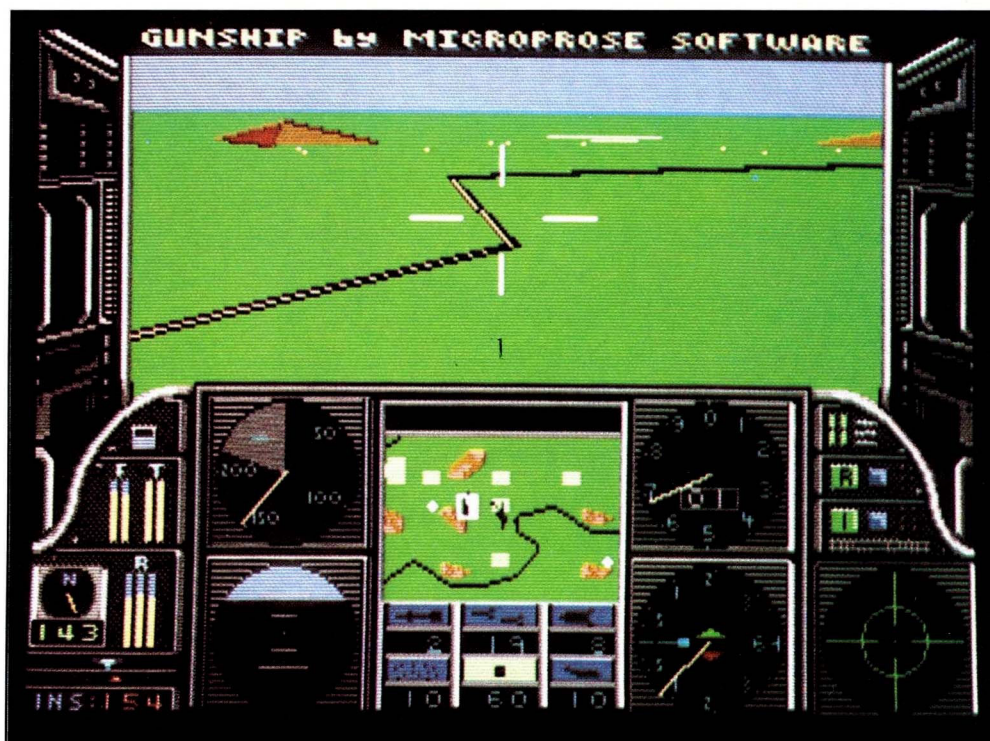
Carregue de manhã o SIDEKICK na memória do computador e fique acompanhado durante todo o dia com esta poderosa ferramenta de trabalho, mesmo utilizando o computador para explorar outro software.

PREÇO: 3 900$00 **REF. 303, postal 3**

MANUAL DE BASIC 2 PARA PC

Ainda não sabe BASIC? Já conhece outro BASIC? Mas não conhece o BASIC 2!
Esta é a linguagem de programação que lhe faz falta conhecer. As sua potencialidades são muitas e convidamo-lo a vir descobri-las.
Através da utilização das janelas do GEM você estabelece um diálogo permanente com a máquina.
O BASIC 2 utiliza, para além de muitas outras particularidades que não encontram nas versões de BASIC disponíveis no mercado, ficheiros indexados próprios das linguagens de gestão. Esta é uma das muitas características que o distingue dos outros.
E, concerteza, muito mais.
Este é o manual que lhe faz falta na sua secretária. Não perca a oportunidade de adquirir o manual ao preço... bem... ao preço AMSTRAD.

PREÇO: 2 690$00
REF. 304, postal 3

A CARTA EGA DO PC 1640

Quando a IBM concebeu o PC, pensou, equivocadamente, que para as aplicações profissionais só se iria necessitar do modo de texto, enquanto que para o uso doméstico faria falta um modo gráfico (para os jogos, claro está).

O simulador de voo Gunship ao ser executado num Amstrad PC 1640 no modo EGA de 640 por 350 pixels e 16 cores.

POR esta razão, e para além do Monochrome Display Adapter (MDA), incluido no PC original, a multinacional americana lançou para o mercado a carta Color Graphics Adapter (CGA). Enquanto que a MDA oferecia textos de alta resolução (caracteres de 9x14 pixels), mas nenhum modo gráfico, a CGA incluia vários modos gráficos, mas uma resolução nos textos muito inferior (caracteres de 8x8 pixels). A solução adoptada pela IBM não satisfez ninguém. Os utilizadores profissionais do PC, apesar do que a IBM queira pensar, necessitavam de gráficos e a CGA, embora os suportasse, tornava-se escassamente adequada para trabalhar em modo texto, já que a ínfima resolução dos seus caracteres 8x8 pontos - os tornava de difícil e incómoda leitura.

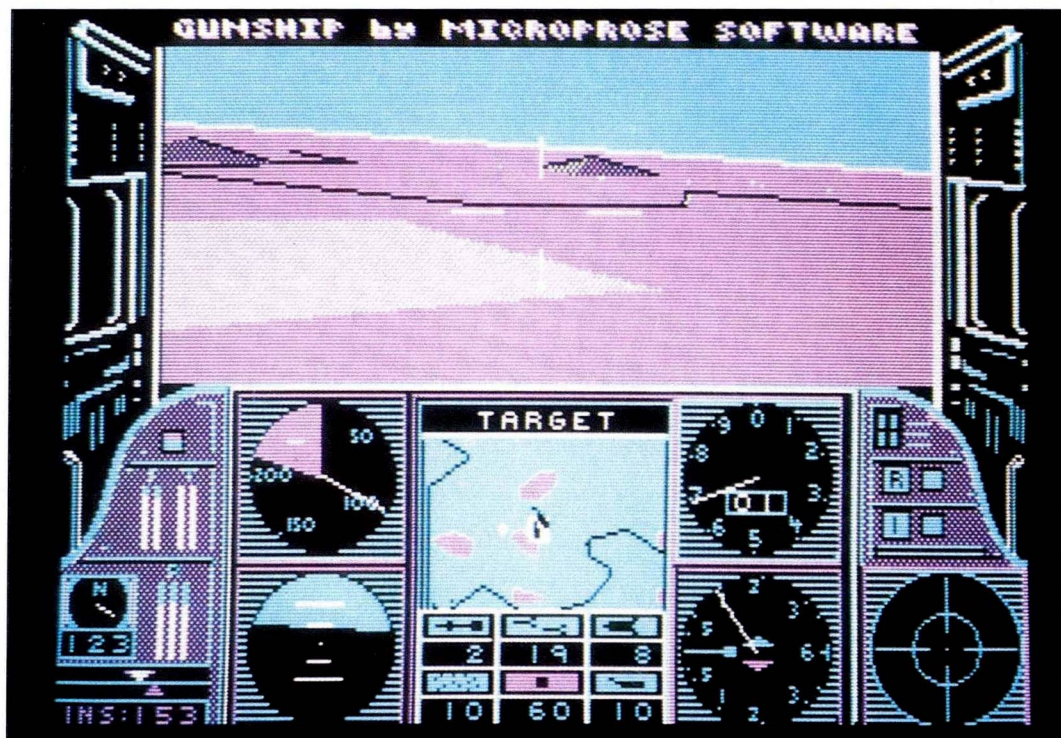
Durante algum tempo muitos utilizadores de PC's trabalharam simultaneamente com dois monitores, um para visionar os textos da MDA e outro para os gráficos da CGA. Entretanto, a firma americana Hercules Computer Technology lançou a famosa carta Hercules, com um modo gráfico monocromático de 720x348 pontos e caracteres de 9x14 pontos. Apesar de ser totalmente independente da IBM, a carta Hercules converteu-se rápidamente no standard profissional, até porque as aplicações mais populares a suportavam caso do Lotus 123, Framework e Symphony.

Neste estado de coisas o "Big Blue" (nome por também é conhecida a IBM), com um atraso de vários anos, apresentou por fim uma carta que combinava uma magnífica resolução de caracteres em modo texto com um modo gráfico bastante mais aceitável que o da CGA. Tratava-se da Enhanced Graphics Adapter, muito mais conhecida pela sigla EGA. Contudo, a sua popularização não foi imediata, e teve que se esperar uma diminuição do seu elevado preço inicial, o que só ocorreu com a aparição das primeiras EGA construidas e comercializadas por fabricantes independentes.

Na actualidade, a carta EGA vai-se impondo como o standard gráfico para os compatíveis PC, embora a desordem inicial e os problemas de incompatibilidade entre as diversas cartas gráficas ainda subsistam. E para cúmulo, a IBM voltou a semear a confusão com a introdução de novos modos gráficos (VGA) nos recentes computadores da gama Personal System/2.

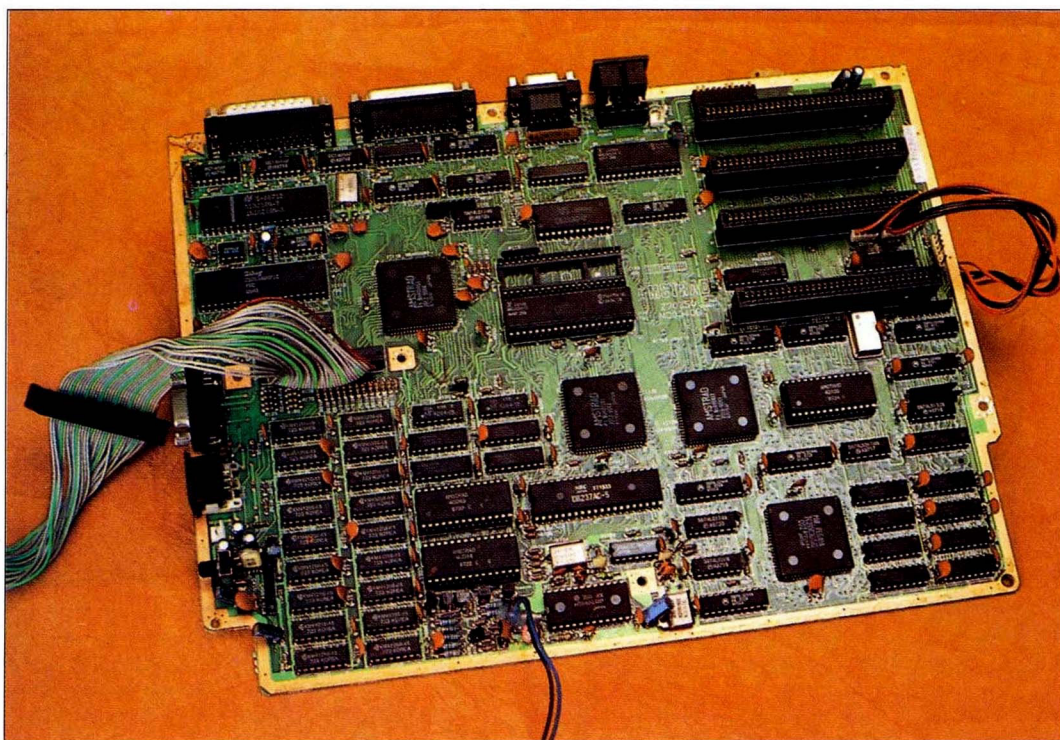
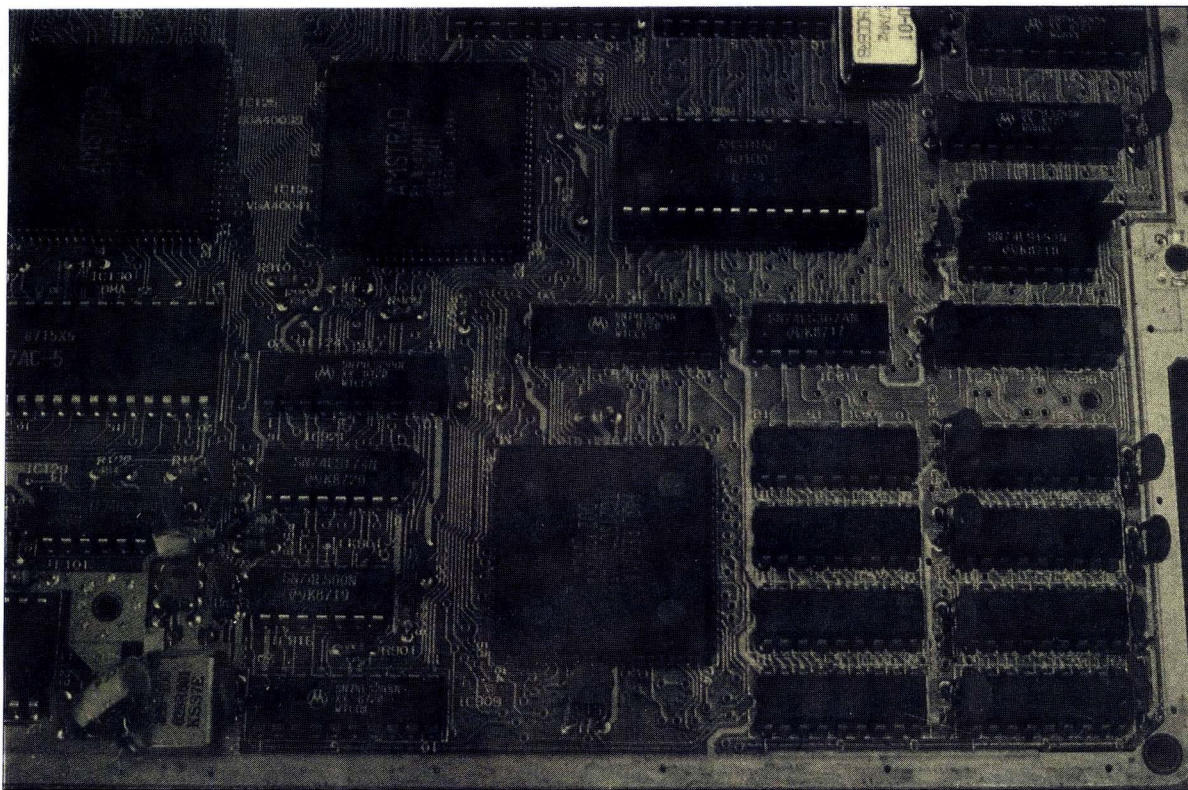
Os modos gráficos da carta EGA

Ante este obscuro panorama, a alternativa adoptada pela Amstrad no PC 1640 não pode ser mais louvável: os circuitos de video do PC 1640 são capazes de emular as cartas EGA, CGA e Hercules, trabalhando em todos e cada um dos modos próprios de qualquer delas. Unicamente o écrã restringe o re-

O mesmo jogo num Amstrad PC 1512. Observe-se bem a diferença.

pertório de modos de visualização possíveis no PC 1640. Assim, se se dispõe do monitor a cores ECD (Enhanced Colour Display), não poderão utilizar-se os gráficos Hercules, mas sim qualquer dos modos EGA e CGA. Por outro lado, é possível desactivar o circuito de video do PC 1640 para instalar uma carta gráfica diferente destas (como, por exemplo, a Professional Graphics Controler, da IBM), algo que se tornava impossível no Amstrad PC 1512.

O número de modos de visualização disponíveis na EGA é consideravelmente superior ao da CGA. Alguns deles (de 0 a 6) têm a finalidade de assegurar a compatibilidade com os programas concebidos para a CGA, enquanto que outros são exclusivos da EGA. Vejamos detalhadamente os modos possíveis com o monitor ECD:

- Modos 0 e 1: Modos de texto de 25 linhas por 40 colunas, com 16 cores. Equivalentes aos modos 1 e 2 da CGA, mas com caracteres de 14x8 pontos e resolução de 320x350 pixels.

- Modos 2 e 3: Como os dois modos

A placa base do PC1640.

Detalhe dos circuitos de video do PC 1640. À direita podem ver-se os chips de memória gráfica.

anteriores, mas com 25 linhas de 40 colunas cada uma. Equivalentes aos modos 2 e 3 da CGA, mas com caracteres de 14x8 pixels e resolução de 640x350 pixels.

- Modos 4 e 5: Trata-se dos mesmos modos gráficos que existem na CGA, de 320x200 pixels e 4 cores.

- Modo 6: Idêntico ao modo 6 da CGA, ou seja, gráficos de 640x200 pixels com duas cores.

- Modo 13: Exclusivo da carta EGA. é um modo gráfico 320x200 pontos, semelhante ao da CGA, mas com 16 cores.

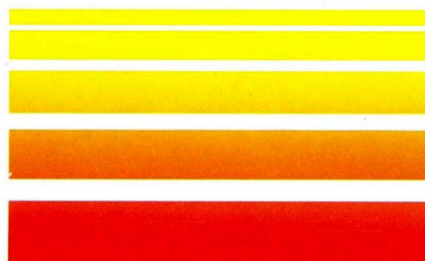
- Modo 14: Exclusivo da carta EGA. É o modo com a máxima resolução gráfica disponível, com 640x350 pixels e 16 cores seleccionáveis duma paleta de 64.

Se a carta EGA for utilizada com um monitor monocromático, passaremos a dispor apenas dos modos 7 e 15. O primeiro deles é um modo de texto, equivalente ao da MDA, com uma resolução de 720x350 pixels, caracteres de 14x9 pixels e écrã de 25 linhas por 80 colunas. O segundo é um modo gráfico de 640x350 pixels, caracteres de 14x8 pixels e 25 linhas por 80 colunas.

E, por último, se se combina a carta EGA com um monitor a cores normal, não do tipo ECD, poderão ser utilizados os mesmos modos que com o monitor ECD, excepto o 16, mas com a importante contrariedade de que a resolução nos modos de texto (0, 1, 2, e 3) será de 320x200 pixels ou de 640x200 pixels, com caracteres de 8x8 pixels.

Utilizando os gráficos EGA

A situação pouco menos que caótica da proliferação de cartas e modos gráficos diferentes é resolvida pelas empresas de software do melhor modo possível. Assim, enquanto certos programas suportam várias cartas gráficas sem mais complicações que as derivadas do processo de instalação do software, algumas aplicações profissionais e, sobretudo, muitos jogos, foram escritos e concebidos para apenas uma carta video (geralmente a CGA), de forma que, mesmo dispondo de um PC 1640 com monitor a cores ECD, o programa será incapaz de tirar partido das prestações da carta EGA. Contudo, embora um programa não utilize os gráficos de alta resolução da carta EGA, sempre que trabalhe num modo de texto mostrará no ecrã caracteres de 8x14 pixels, o que supõe uma importante melhoria relativamente aos 8x8 pixels da carta CGA.

A situação é menos complexa nos programas que se executam sobre o ambiente gráfico GEM, capazes de se ajustarem automaticamente ao modo gráfico que se está a utilizar num dado momento. Assim, os utilizadores do novo PC 1640 ECD observarão que o aspecto do GEM Desktop sobre o seu computador é bastante mais atractivo que no PC 1512. Os ícones, embora de menor tamanho, são muito mais nítidos e no texto apenas podem apreciar-se os pontos que compõem os caracteres. No caso do GEM Paint, ao contrário do que ocorre nos programas de desenho não baseados no GEM, os écrãs criados no modo de alta resolução especial do PC 1512 poderão ser carregados sem dificuldade no PC 1640, adaptando-se automáticamente à resolução de 350 linhas verticais.

DEBUG

Muito mais que um disassemblador

Na disquete do sistema operativo MS-DOS encontra-se um programa completamente ignorado nos manuais do PC 1512. Trata-se do Debug, um potente monitor-disassemblador de código máquina.

Do mesmo modo que alguns utilizadores dos CPC 464 e 6128 utilizam disassembladores na busca de pokes para os jogos mais conhecidos, os possuidores dos PC's da Amstrad podem utilizar o Debug para analisar o funcionamento do sistema operativo ou para localizar e sanar os pequenos erros (bugs) dos seus programas em código máquina.

A tarefa realizada pelo Debug consiste basicamente em explorar a memória dos PC's, formada por numerosas células, cada uma das quais adopta um valor numérico determinado. O Debug, como qualquer disassemblador é capaz de traduzir estes valores em instruções de linguagem assembler e, como é lógico, de modificar o conteúdo da memória, ir à procura de uma cadeia de caracteres determinada, mudar blocos de memória de posição, carregar e gravar ficheiros ou sectores do disco e executar programas passo a passo, de maneira controlada e mostrando os estados dos registos do processador em cada momento, característica esta a que deve o nome de monitor de código máquina. Ou seja, proporciona todos os meios necessários para a depuração de programas, incluindo ainda um modesto assemblador.

Para começar a trabalhar com o Debug não é necessário, embora pareça o contrário, dominar o código máquina. Uma das suas aplicações mais interessantes e ao mesmo tempo mais divertidas, é a modificação de programas, personalizando-os, corrigindo bugs ou adaptando-os ao nosso equipamento. E para isto só são requeridos conhecimentos rudimentares do funcionamento do PC.

O sistema de funcionamento hexadecimal

A linguagem materna dos PC's, ou seja, aquela que os PC's entendem sem necessidade de intérpretes e compiladores que a traduzam, é o código máquina. As suas instruções consistem em números binários de oito cifras e são totalmente enigmáticas para um ser humano normal. Para evitar estas intermináveis cadeias de 'zeros e uns', trabalha-se normalmente em base hexadecimal, já que os números hexadecimais se convertem rapidamente em binários.

Uma das características do processador Intel 8086, que é preciso conhecer antes de utilizar o programa Debug, é o seu peculiar sistema de endereçamento da memória. Embora o seu bus de endereços de 20 bits o capacite para endereçar até 1 Mb, os seus registos internos são de 16 bits e permitem referenciar unicamente um máximo de 64 Kbytes. Para superar esta restrição utiliza-se o procedimento conhecido por segmentação que, como o seu próprio nome indica, consiste em dividir a memória em segmentos de 64 Kb. O endereço de começo dos quatro segmentos com os quais trabalha o microprocessador (códigos, dados, stack e extra) guardam-se nos quatro registos de segmento. Os endereços de memória calculam-se automaticamente, multiplicando por 16 o conteúdo do registo de segmento apropriado e somando ao resultado um valor de deslocação, que pode ser uma constante ou uma quan-

O Debug suporta a sintaxe standard do assemblador dos processadores 8086/8088, assim como o jogo de instruções do coprocessador aritmético 8087.

COMANDO	FUNÇÃO	SINTAXE
Assemble	Asssembler de código máquina	A [endereço]
Dump	Cópia do conteúdo da memória	D[endereço] ou D[limite]
Enter	Substitui o conteúdo de um ou mais bytes	E endereço [listagem]
Go	Executa com pontos de ruptura opcionais	G [=endereço]
Hexarithmetic	Soma e subtrai números hexadecimais	H valor valor
Move	Transfere blocos de memória	M limite endereço
Quit	Termina a execução do Debug	Q
Register	Mostra o conteúdo dos registos	R [nome do registo]
Search	Procura caracteres	S [limite listagem]
Trace	Executa passo a passo	T [=endereço] [valor]
Unassemble	Disassembla instruções	U [endereço]
Write	Grava ficheiros ou sectores de disquete	W [endereço [unidade sector sector]]

Resumo dos principais comandos do Debug. As convenções de sintaxe assumidas são equivalentes às que se utilizam no manual dos PC's Amstrad.

tidade armazenada noutro registo.

O Debug utiliza esta convenção e, por conseguinte, quando seja necessário passar-lhe um endereço teremos que indicar o começo do segmento e o deslocamento ou distância a que se encontra o mesmo. Por exemplo, D010:AABB é uma das formas em que se pode representar a posição de memória DABBB. D010 marca o começo de um segmento que se estende do endereço D0100 (o registo de segmento multiplicado por 16) ao E0100, enquanto que o valor de deslocamento AABB assinala a posição em que se encontra a uma distância de AABB bytes do começo do segmento.

Um exemplo de trabalho com o Debug

Como exemplo das possibilidades básicas do Debug vamos descrever um procedimento simples para mudar o nome a um dos comandos do sistema operativo.

O MS-DOS dispõe de duas ordens diferentes para apagar os ficheiros: DEL e ERASE. O nosso propósito é utilizar em lugar desta última, a expressão portuguesa APAGA. Para isto temos que levar em conta que ERASE é um comando interno do MS-DOS, isto é, não se trata de um comando externo, como FORMAT ou DISKCOPY, mas sim de um que é controlado directamente pelo processador de comandos COMMAND.COM. Em consequência, o primeiro passo será carrregar o COMMAND.COM a partir do Debug, o que se consegue introduzindo o comando DEBUG COMMAND.COM. Uma vez chegados a este ponto, convém advertir que devemos trabalhar sempre com uma cópia (obtida a partir do DISKCOPY) da disquete que contém o MS-DOS; de outro modo poderíamos perder irremediavelmente o sistema operativo.

Uma vez no Debug, desaparecerá o indutor do sistema e o ecrã mostrará como indicativo um traço. O primeiro comando do Debug que iremos utilizar é R (Register), cuja função é analisar o conteúdo hexadecimal de todos os registos do 8086. O registo CX indica o tamanho do programa, que, no caso do COMMAND.COM é de 5C3Ch bytes. DS, ES, SS e CS devolvem a posição de origem dos quatro segmentos de memória, enquanto que o ponteiro de instruções IP mostra o endereço da primeira instrução a executar, que se encontra, como todos os programas de terminação .COM, a 100h bytes do começo do segmento de código.

A seguir utilizamos o comando S (Search), que procura na memória uma dada cadeia de caracteres. A sua sintaxe é S limites listagem, e a listagem pode introduzir-se pela forma de caracteres ASCII ou directamente em hexade-

```
C:\>debug
-a 100
2CEF:0100 MOV DX,010A
2CEF:0103 MOV AH,9
2CEF:0105 INT 21
2CEF:0107 INT 20
2CEF:0109 NOP
2CEF:010A DB 0A 0D "AMSTRAD MAGAZINE" 0A 0D "$"
2CEF:011F
-Ndemo.com
-RCX
CX 0000
:1F
-W
Writing 001F bytes
-G

AMSTRAD MAGAZINE

Program terminated normally
-Q

C:\>DEMO

AMSTRAD MAGAZINE

C:\>
```

Utilização do Debug como assemblador de código máquina.

cimal. No nosso caso vamos digitar S 100 L 5C3C "ERASE", para indicar ao Debug que localize a cadeia de caracteres ASCIII "ERASE" na zona de memória compreendida entre as posições 0100h e 5D3C, ou seja, em toda a memória ocupada por COMMAND.COM. O endereço obtido é o 2DDD:4EC3, que corresponde ao quadro de comandos do COMMAND.COM. A cópia dos endereços de memória contíguos a 4EC3 com o comando D4EA0 mostrará alguns dos comandos armazenados no quadro, como DIR, RENAME, TYPE, COPY e outros.

O procedimento para substituir ERASE por APAGA é tão simples como digitar E 4EC3 "APAGA", que introduz a nova cadeia ASCII na posição ocupada anteriormente por ERASE.

A gravação em disquete da versão modificada de COMMAND.COM consegue-se com o comando W (Write), que apaga a versão primitiva, motivo porque importa não trabalhar com o original de nenhum programa. Para finalizar, abandonamos o programa Debug introduzindo o comando Q (Quit), voltando assim ao sistema operativo. Se agora inicializarmos o PC 1512/1640 fazendo-o arrancar com a disquete-cópia do MS-DOS que estivemos a utilizar, comprovaremos que se podem apagar ficheiros com o comando APAGA, cuja sintaxe é idêntica à do ERASE. Este procedimento pode utilizar-se para modificar à nossa vontade o nome de qualquer comando interno do MS-DOS.

Debug como assembler

Para além das suas possibilidades como monitor e disassemblador de código máquina, o Debug tem a capacidade de assemblar instruções de linguagem assembler directamente da memória. Suporta a sintaxe standard do assembler 8086/8088, assim como o jogo de instruções do coprocessador aritmético 8087.

Como exemplo do funcionamento do Debug como assemblador descreveremos passo a passo a criação de um breve programa em código máquina, que imprime no ecrã dos PC's Amstrad o nome da nossa revista.

Em primeiro lugar carregamos o Debug sem especificar nenhum nome de ficheiro. Digitamos A seguido de Enter e entramos no modo assemblador. O PC mostra nesse momento um endereço de memória (XXXX:0100) e permanece à espera de instruções. Escrevemos agora cada uma das seguintes palavras, sem digitar os comentários (entre parêntesis):

MOV DX, 10A <Enter> (carrega DX com o endereço do começo do texto)

MOV AH,9 <Enter> (chama a função do DOS encarregada da impressão de caracteres no ecrã)

INT 21 <Enter>

INT 20 <Enter> (interrupção do final do programa)

NOP <Enter> (instrução nula)

DB 0A 0D "AMSTRAD MAGAZINE" 0A 0D "$" <Enter> (texto a imprimir) <Enter>

Ndemo.com <Enter> (dá um nome ao ficheiro)

RCX <Enter>

1F <Enter> (carrega CX com o comprimento do programa)

W <Enter> (grava o programa em disco)

Q <Enter> (termina a execução do Debug)

Ao concluir este processo teremos criado no disco um ficheiro com o nome de DEMO.COM, que ao executar-se imprime a frase AMSTRAD MAGAZINE.

O Debug dispõe de muitas outras possibilidades que seria impossível analisar neste artigo. Felizmente, existe uma extensa relação de livros dedicados ao sisterma operativo MS-DOS, nos quais se descrevem minunciosamente estes programas.

O 8086 é capaz de endereçar até um megabyte de memória, utilizando uma técnica conhecida como segmentação.

```
A:\>debug command.com
-r
AX=0000  BX=0000  CX=5C3C  DX=0000  SP=FFFE  BP=0000  SI=0000  DI=0000
DS=2DDD  ES=2DDD  SS=2DDD  CS=2DDD  IP=0100   NV UP EI PL NZ NA PO NC
2DDD:0100 E9DD0B        JMP     0CE0
-s 100 l 5c3c "ERASE"
2DDD:4EC3
-d 4ea0
2DDD:4EA0  08 05 45 58 49 53 54 87-08 00 03 44 49 52 03 0D   ..EXIST....DIR..
2DDD:4EB0  0D 06 52 45 4E 41 4D 45-01 3C 10 03 52 45 4E 01   ..RENAME.<..REN.
2DDD:4EC0  3C 10 05 45 52 41 53 45-01 E4 0F 03 44 45 4C 01   <..ERASE....DEL.
2DDD:4ED0  E4 0F 04 54 59 50 45 01-B4 10 03 52 45 4D 02 04   ...TYPE....REM..
2DDD:4EE0  01 04 43 4F 50 59 03 01-27 05 50 41 55 53 45 02   ..COPY..'.PAUSE.
2DDD:4EF0  D7 0F 04 44 41 54 45 02-5B 1E 04 54 49 4D 45 00   ...DATE.[..TIME.
2DDD:4F00  5E 1F 03 56 45 52 00 AA-11 03 56 4F 4C 01 54 11   ^..VER....VOL.T.
2DDD:4F10  02 43 44 01 61 16 05 43-48 44 49 52 01 61 16 02   .CD.a..CHDIR.a..
-e 4ec3 "APAGA"
-d 4ea0
2DDD:4EA0  08 05 45 58 49 53 54 87-08 00 03 44 49 52 03 0D   ..EXIST....DIR..
2DDD:4EB0  0D 06 52 45 4E 41 4D 45-01 3C 10 03 52 45 4E 01   ..RENAME.<..REN.
2DDD:4EC0  3C 10 05 41 50 41 47 41-01 E4 0F 03 44 45 4C 01   <..APAGA....DEL.
2DDD:4ED0  E4 0F 04 54 59 50 45 01-B4 10 03 52 45 4D 02 04   ...TYPE....REM..
2DDD:4EE0  01 04 43 4F 50 59 03 01-27 05 50 41 55 53 45 02   ..COPY..'.PAUSE.
2DDD:4EF0  D7 0F 04 44 41 54 45 02-5B 1E 04 54 49 4D 45 00   ...DATE.[..TIME.
2DDD:4F00  5E 1F 03 56 45 52 00 AA-11 03 56 4F 4C 01 54 11   ^..VER....VOL.T.
2DDD:4F10  02 43 44 01 61 16 05 43-48 44 49 52 01 61 16 02   .CD.a..CHDIR.a..
-w
Writing 5C3C bytes
```

Listagem de uma sessão de trabalho com Debug.

PPC ORGANISER

A SECRETÁRIA ELECTRÓNICA

Os computadores portáteis Amstrad PPC, agora lançados em Portugal, vêm acompanhados por um programa utilitário muito interessante, o PPC-Organiser. Produzido pela Clasma Software sob a designação de "A La Carte". Tem sido comparado ao "SideKick" e a programas similares, mas a semelhança é só aparente. Na verdade dispõe dos elementos clássicos de "secretária electrónica", mas é diferente no sentido de que não deixa ao utilizador o trabalho de coordenar as funções daqueles. O "Organiser" coordena tudo, organiza tudo. É uma verdadeira secretária electrónica.

O programa tem por núcleo uma base de dados, que reúne os elementos introduzidos no bloco de apontamentos — endereços, números de telefones, ocupação ou cargo, e muitas coisas mais, desde modos pessoais de tratamento a curtas descrições de equipamento — em matrizes com 12 linhas de 40 caracteres. Um processador de texto mais completo do que o habitual pois dispõe de "bold", sublinhado, de corte e deslocação, busca e substituição de blocos de texto, permite escrever cartas e outros documentos, e pode trabalhar em conjunto com uma variedade de matrizes de cartas, facturas, avisos, etc. Nos documentos assim criados podem ser integrados os elementos constantes da base de dados. Vai muito além do "mail merge". Por exemplo: pode perguntar-se ao programa por uma palavra ou uma frase, e ele mostra todos os documentos onde ela ocorrer. Se não encontrar nada do que foi pedido, pergunta se se deseja o que houver de mais próximo...e ele procura-o! A busca pode também ser referida à data e hora, mas deve notar-se que o relógio-calendário dos Amstrad PPC não dispõe de alimentação independente.

A calculadora é também melhor que o normal. Trabalha nas bases décimal, hexadécimal e binário, e no primeiro caso permite fixar o número de algarismos à esquerda e direita da vírgula.

Como se disse, o que mais impressiona é a integração dos vários elementos — pode passar-se toda a espécie de informação de uns para outros, fazer contas na calculadora, transferir o resultado para o bloco de apontamentos e vê-lo integrado a seguir, automaticamente, num documento. O programa é tão "inteligente" que até mantém os registos das cartas enviadas para cada pessoa constante das listas de endereços, e mostra-os, se necessário.

Tudo muito prático, portanto — excepto alguns problemas. As revistas britânicas dizem que o programa é residente, isto é, fica carregado na memória e pode ser chamado quando necessário. Por mais que procurássemos fazer isso não o conseguimos, nem obtivémos informação de como é feito. O programa com que estamos a trabalhar não veio acompanhado por quaisquer instruções e as distribuídas na Grã-Bretanha com os protótipos reduziam-se a um folheto muito elementar, aparentemente provisório. É possível que essa modalidade tenha sido abandonada, pois exigia um disco de memória de 220 kb o que por certo era excessivo — a disquete do sistema cria um de somente 100 kb e não dispõe do programa NVR para criar o seu valor.

Em consequência, o PPC Organiser pede constantemente elementos à respectiva disquete, quase sempre para o "Help", que é muito cmpleto e acompanha todas as operações, numa janela na parte inferior da imagem. Infelizmente isso implica demoras e um consumo adicional de energia que pesa muito na duração das pilhas que alimentam os Amstrad PPC, pelo que o seu uso se deve limitar às ocasiões em que o computador esteja a ser alimentado pela rede. Seria interessante ver o Organiser trabalhar num disco rígido, mas os PPCs ainda não o têm e o programa foi concebido para ser usado exclusivamente neles.

Eurico da Fonseca

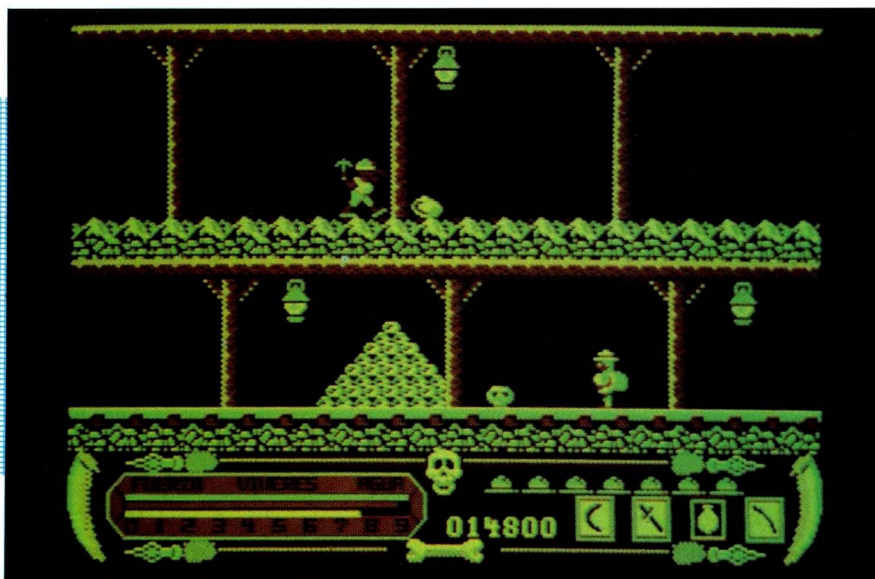
DR. LIVINGSTONE, SUPONHO

A aventura está servida para os utilizadores do PC. Pela mão da Opera Soft irão viver uma emocionante busca em pleno coração da África Negra.

Os tranquilos utilizadores dos PC Amstrad, pouco habituados às emoções fortes, têm neste programa um dos primeiros "arcades" puros e autênticos para o seu computador, para além de ser um dos primeiros jogos desenvolvidos para os compatíveis PC por uma casa de software espanhola, Opera Soft.

O enredo do jogo não é nem mais nem menos do que a apaixonante aventura que viveu o jornalista Sir H. Morton Stanley, do New York Herald, em busca do professor Livingstone através do continente africano, em plena época colonial. A sua finalidade era conseguir o exclusivo do descobrimento das origens do Nilo, tarefa de que foi encarregado pelo seu jornal; a nossa missão será mais simples, já que só temos de encontrar o professor, do qual só sabemos que se encontra nalgum lugar da intrincada selva africana. Mas para o conseguir teremos de percorrer inúmeros cenários, saltando ou disparando qualquer das nossas três armas: boomerangs, machetes e granadas de mão, afim de acabar com o grande número de "alimárias" que povoam habitualmente as selvas.

Os gráficos do programa são algo digno de ser visto, com uns movimentos quase perfeitos, tanto na personagem principal (que veremos a baixar-se, a saltar, apanhar a vara ou aturdir-se) como nos inimigos, que farão praticamente de tudo, desde disparar dardos, balas e, inclusivamente cocos até nos raptar em voo e nos abandonar em lugares inacessíveis donde teremos que começar novamente a busca; passando, claro está, por perigosas plantas carnívoras a que serviremos de pequeno-almoço no caso de nos aproximar-

mos demasiado.

Pessoalmente, o programa recorda-nos as famosas aventuras das máquinas recreativas, donde nunca conseguiamos passar do segundo ecrã, quer por falta de recursos monetários, quer por falta de habilidade. Existe agora a hipótese de vingança, embora, claro está, iremos sofrer das limitações da nossa máquina, ou seja, do facto de alguns gráficos serem esquemáticos e algo repetitivos, factos que normalmente nos passarão despercebidos pela quantidade e variedade de lagos, piranhas, águias, indios e muitos outros espécimens que nos farão divertir e desesperar ao mesmo tempo. O jogo vai subindo de dificuldade, através da superação de obstáculos cada vez mais difíceis, em forma de acidentes naturais e inimigos que nos espiam constantemente e que iremos descobrindo à medida que avançarmos no jogo. É o caso de povoações indias, minas, o grande templo onde uma lindíssima nativa nos oferecerá uma dança sensual a troco, habitual nas mulheres, de cinco gemas que deveríamos ter recolhido antes.

E a seguir a este momento de descanso teremos que retornar às nossas ocupações de exploradores, entre as quais se inclui o dever de manter o nosso reporter, Stanley, muito bem alimentado através duma apanha de provisões que iremos encontrando no caminho. Em caso contrário, a nossa personagem perecerá vitima da fome ou da sede, isto supondo que se safa das numerosas armas dos indigenas ou das intermináveis investidas da fauna e da flora típica do lugar.

Quanto ao som, segue a linha exótica do programa, embora talvez seja a única coisa que destoa com os restantes aspectos, relativamente à qualidade - mais pelas limitações impostas pelo computador que pelo trabalho dos programadores, o qual é realmente bom.

Como provavelmente já suspeitaram, o jogo é absorvente a cem por cento, facto que se costuma dar quando se junta aventura, bons gráficos e dificuldade. Tem que se levar em conta, portanto, com uma certa filosofia, para nos habituarmos à ideia de que não vamos conseguir terminá-lo no mesmo dia da sua compra. A Opera Soft conseguiu assim demonstrar que o PC Amstrad não só serve para jogar como se torna ideal para isso.

Distribuidor: Opera Soft (Espanha)
O melhor: A variedade e a qualidade dos gráficos.
O pior: A dificuldade, nalguns casos exçessiva.

UM PAR DE "PASCAIS"

A Fonte

Na linguagem de compilador, a sua versão do programa (normalmente um ficheiro de texto ASCII) é chamada de Programa Fonte (Source Program), enquanto que a versão em código máquina produzida pelo compilador se chama Progama Objecto (Object Program). Alguns compiladores produzem uma versão intermédia que só pode ser executada depois de 'ligada' a outras partes da linguagem, fornecida com o compilador. Provocando alguma confusão, estes ficheiros intermédios são por vezes chamados de Formato Objecto (Object Format), com uma extensão .OB no nome. No entanto, os únicos ficheiros contendo programas directamente executáveis sob MS-DOS são os que incluem nomes com extensões .COM ou .EXE.

QUANDO você compra um PC as probabilidades são que ele venha com um interpretador BASIC, apesar de existir um mercado crescente de linguagens alternativas. A mais popular é sem dúvida o Pascal, particularmente o Turbo Pascal da Borland, com mais de 500.000 cópias vendidas em todo o mundo.

O Pascal é uma linguagem estruturada de aplicação geral, muito apreciada pelas universidades e considerada ideal para a aprendizagem de programação. Em muitos aspectos é similar ao BASIC, mas enquanto o BASIC deriva do FORTRAN, uma linguagem científica concebida para escrever programas curtos sem prestar muita atenção à facilidade de manutenção desses programas, o Pascal tem as suas raízes no ALGOL - também científica, mas concebida para escrever programas muito maiores e onde a simplicidade é vital.

COMPILAÇÃO

O Pascal é normalmente uma linguagem compilada. Isto significa que o programa fonte é traduzido para linguagem máquina, capaz de correr directamente a partir do PROMPT A>. Um interpretador, por outro lado, descodifica o programa cada vez que este corre, necessitando portanto de ter a linguagem de programação sempre carregada. Isto ocupa espaço no disco e na RAM, e é também muito menos eficiente.

Portanto, programas compilados possuem uma elevada portabilidade e correm mais depressa que os interpretados. No entanto, o desenvolvimento do programa é mais difícil, devido à impossibilidade, na maioria dos casos, de ver o conteúdo das variaveis ou a execução no monitor durante os testes, e mais lenta, já que é necessário recompilar o programa inteiro depois de cada emenda.

ESTRUTURA

O Pascal é uma linguagem estruturada, a dois níveis distintos: controle de execução e modularidade.

O controle da execução respeita à ordem e escolha de quais os comandos a executar quando o programa corre. O Pascal apresenta um conjunto de comandos condicionais que expressam o objectivo do LOOP ou BRANCH, sendo o mais simples **IF...ELSE**, que pode ser usado para executar secções completas do programa condicionalmente. Isto, ao contrário da maior parte dos BASIC's, que nos obrigam a utilizar GOTO's. Veja os exemplos seguintes:

Pascal:

```
if balance > credit then
begin
  if debtage > 30 then
    stopcredit(accnum);
  printletter(accnum);
  overbal   := overbal + accbal;
  overcount := overcount + 1;
end
else
begin
  okbal     := okbal + accbal;
  okcount   := okcount +1;
end;

10 if balance > credit THEN GOTO 50
20  okbal     = okbal + accbal
30  okcount   = okcount +1
40  GOTO 100
50 REM condicao 'else'
60 IF debtage > 30 THEN GOSUB 1000
70  GOSUB 2000
80  overbal    = overbal + accbal
90  overcount  = overcount + 1
100 REM seccao seguinte do programa
1000 REM inicio da sub-rotina 'fim de credito'
2000 REM inicio da sub-rotina 'letra'
```

No Pascal, a condição controla os comandos que a seguem sem qualquer ambiguidade (entre **begin...end**, a forma de definir um bloco de programa em Pascal). No BASIC, o GOTO separa os comandos da condição e introduz a possibilidade de erros no número de linha de destino. A listagem do Pascal permite igualmente agrupar os IF's de uma forma muito mais legível que a maioria dos BASIC's.

```
      Line 39    Col 1    Insert    Indent  C:COLOR.PAS

type
  AnyString = string[40];

procedure Check;     (Verifica se continua a execucao do programa)
var
  Ch: char;
begin
  Write('Este programa so funciona se possuir uma');
  WriteLn('carta grafica instalada');
  Write('Deseja continuar S/N');
  repeat
    Read(Kbd,Ch)
  until UpCase(Ch) in ['S','N', ?27];
  if UpCase(Ch) in ['N', ?27] then
    Halt;
end;

procedure PaletteDemo;
var
  Ch                          : char;
  PaletteNumber, Background   : integer;
  PaletteChange               : boolean;
```

O Pascal tem também facilidades de looping - **while, repeat...until e for...to**, juntamente com o agrupamento dos comandos **if**. Existe também a construção **case** - uma forma sofisticada de **if...else...if.** O efeito de tudo isto é que, por muito complicados que sejam os objectivos do programador, pode sempre esprimi-los de uma forma tal que o programa fonte apresenta com clareza a lógica do programa.

PROCEDIMENTOS

O exemplo mostra igualmente o uso de um procedimento com um parâmetro em Pascal - **stopcredit (accnum)**. Isto tem diferenças importantes em relação ao **GOSUB** do BASIC. O próprio parâmetro (accnum) tem dois propósitos - mostrar que variaveis são retiradas do programa principal para serem processadas pelo procedimento, e permitir que o procedimento processe diferentes valores cada vez que é usado.

O **GOSUB** do BASIC não atinge nehum destes objectivos. A variável **accnum** não é mencionada em parte alguma, e portanto não existe nenhuma pista para seja o que for usado para controlar a subrotina, e a subrotina terá que referenciar **accnum** directamente pelo nome, tornando impossível a sua utilização noutras partes do programa para processar o conteúdo de outras variaveis.

Os procedimentos são a principal ferramenta da programação modular. Qualquer deles é, com efeito, um pequeno programa em si e utiliza as suas próprias variáveis - e assim podemos escreve-los e testá-los independentemente de outras partes do programa, e podem ser usados por qualquer programa que deles necessite. Os procedimentos permitem igualmente a simulação - o exemplo que nós demos pode ser escrito e testado utilizando uma versão simulada do procedimento stopcredit antes de ser escrita a definitiva.

TIPOS DE DADOS

Todas as linguagens apresentam vários tipos de dados, como os numéricos, string e boleanos, quaisquer deles com uma gama de valores possíveis (-32768 até +32767 para um inteiro, por exemplo). Em muitos casos isto é muito mais do que os valores que uma variável realmente necessita.

OS PONTOS NEGATIVOS

O Pascal é rápido, estruturado e tem inúmeras possibilidades - então onde é que estão os problemas? Bom, é um pouco fraco em gestão de ficheiros - melhor que o BASIC, visto que gere os registos directamente, mas sem indexação. Da mesma forma, a gestão de ecrãns é limitada, sem possibilidades de janelas ou form-filling. Estes (e outros) problemas costumam ser resolvidos por livrarias externas ou procedimentos pré-escritos para os vários compiladores. Mas isto provoca outro tipo de problema: os standards.

Apesar de ser um standard internacional, o Pascal tem inúmeras versões, motivadas pela relativa falta de performance da linguagem original. Está tudo bem se você se mantiver "agarrado" a um determinado complilador, mas se não, começarão a aparecer dificuldades.

Provavelmente a característica pior do Pascal é a necessidade de colocar todas as definições de procedimentos no programa fonte antes de eles serem chamados - uma tremenda gaffe que significa que se concebeu o seu programa "de cima para baixo", terá que o escrever "de baixo para cima".

Finalmente, o Pascal pode parecer irritantemente criptíco e pomposo. Se nunca ouviu falar antes de "Enumerated Scalar Types" ou "Set Disjunction", então aprendê-lo poderá ser uma tarefa árdua - trata-se de uma linguagem académica muito virada para a teoria da programação.

Mas, no seu conjunto, o Pascal tem mais pontos fortes que fracos - é uma boa linguagem para tudo desde um sistema de controle de stocks (com livrarias de procedimentos adequados) até à concepção do sistema operativo do Macintosh. Com mais de meio milhão de programadores em todo o mundo, não podem estar todos errados!

Utilitários Borland

A Borland desenvolveu um conjunto de utilitários para serem utilizados com o Turbo Pascal - base de dados, gráficos, editor e jogos. Cada um deles consiste em procedimentos Pascal para integrar nos seus programas, com um manual que os explica. Nos próximos números estaremos atentos a estes desenvolvimentos.

TURBO PASCAL

Este produto da Borland International, lançado em 1983, quebrou definitivamente um mito que vigorava nessa altura: o de que só existiam duas espécies de programadores - os que utilizavam interpretadores, que eram considerados próprios apenas para computadores domésticos, e os outros, aqueles que atravessavam uma selva de editores de fontes, compiladores multipass, livrarias,etc. e que faziam jus à ideia que compiladores eram para homens a sério.

A ideia por detrás do Turbo Pascal era fazer um sistema de desenvolvimento de aplicações o mais parecido possível com um interpretador (edição imediata de código de fonte depois de erros de execução, execução imediata do programa depois da edição) mantendo ao mesmo tempo as virtudes de um compilador - baseado em código objecto com tempos de execução rápidos). E, em grande parte, conseguiu esses objectivos.

O ambiente Turbo consiste num editor de texto, um compilador de Pascal, e um mecanismo de teste. A partir do menu central é possível aceder ao ficheiro fonte, compilar o programa, corrê-lo, descobrir os bugs, voltar atrás e começar de novo - o clássico estilo "editar-compilar-testar". O resultado final é um ficheiro .COM que pode ser executado a partir do PROMPT A>.

Mini Pecan Pascal
A Pecan acaba de desenvolver o PDQ Pascal, uma versão condensada do Pecan Pascal a um preço bastante mais barato. Contém um editor, suportado em memória, e um manual, com uma secção de aprendizagem. Para estudantes ou compradores de mais de 20 versões o preço é ainda mais reduzido.

MENU

O menu de abertura parece surpreendentemente parco - apenas algumas opções e um cursor. O Turbo Pascal não tem capacidades como **help screens, pull-down menus ou janelas** mostrando-se um pouco antiquado neste aspecto.

O editor é baseado na memória, o que o torna bastante rápido mas limita o tamanho do programa a editar, ainda que 61 Kb seja já bastante. Utiliza o mesmo tipo de teclas de comando [Ctrl] do Wordstar, para mover blocos e outras acções semelhantes, ainda que sem a ajuda dos menus tipo Wordstar seja necessário um certo esforço de memória. O menu inclui o search-replace, read-write, funções de disco, etc., mas não comporta o multi-file edit, que seria util para recuperar pequenos blocos de programação de outros programas.

O compilador é bastante rápido - mesmo um programa de 20 páginas não leva mais de um minuto. Tem a possibilidade de integrar o editor, quando é encontrado algum erro. Nesse caso, o Turbo Pascal apresenta imediatamente o ecrã de editar, com o cursor posicionado na instrução errada, bastando apenas modificá-la.

Esta integração extende-se, embora de outra forma, ao teste do programa. Se um erro de 'run-time' ocorre, a mensagem inclui um endereço em memória que pode ser integrado no editor, buscando automáticamente a linha de programa a que corresponde - um bom compromisso que cobre parcialmente a falta de capacidades de debug, existentes noutros compiladores mais caros.

LINGUAGEM

Ao princípio, o Turbo tinha o handicap de não ser um Pascal 'standard' - mas com todos os utilizadores que tem hoje em dia já ganhou ele próprio o estatuto de standard. Existem no entanto algumas diferenças e capacidades adicionais, particularmente ao nível da compatibilidade IBM PC. Mas não há qualquer dúvida que aprendendo esta linguagem se está a aprender Pascal, ainda que sejam necessários alguns ajustamentos quando se trabalhar com outras implementações.

As novas capacidades do Turbo no âmbito da compatibilidade PC dizem principalmente respeito à apresentação no ecrã. É possível definir janelas, selecionar as cores do ecrã, desenhar gráficos e um conjunto de comandos tipo 'Turtle' substituem instruções como 'turn left' e 'go backwards'.

O modo IBM CGA do PC 1512 é igualmente suportado (três cores), mas não o modo especial de 16 cores da Amstrad, ou o modo EGA disponível no PC 1640. Existe também um comando de som que emite uma dada frequência até ser desligado - muito útil uma vez que é difícil tirar algum som decente de um PC.

Estão também disponíveis funções para leitura directa do teclado, criar e modificar directorias MS-DOS, retorno ao DOS, e comandos de parâmetros de linha, entre outras facilidades como criação de números aleatórios, control de ecrã texto (insert/delete line), e funções integradas como Cos, Sqrt e Round. Em resumo, é uma package bastante útil que oferece o tipo de capacidades específicas que se esperaria encontrar num BASIC integrado.

LIMITAÇÕES DE MEMÓRIA

Um dos contras do Turbo é que só pode produzir ficheiros .COM, com até 64 Kb de memória, incluindo já os 10 Kb de 'runtimes' integrados em qualquer programa. Como compensação existem facilidades de criação de ficheiros de disco de procedimentos compilados, que são automáticamente copiados para uma área de memória comum, à medida que vai sendo necessário. Isto tem as suas limitações, no entanto, e parece ilógico não se poderem correr programas maiores num sistema de 512 Kb.

Por outro lado, com um pouco de esforço é possível fazer uso da toda a memória disponível para armazenamento de dados, ainda que a maior variável única não possa ter mais de 64 Kb. O Turbo Pascal é apresentado numa disquete copiável e corre facilmente num sistema de drive única e 256 Kb de RAM. O manual de 380 páginas é excelente - cheio de exemplos e cobrindo todos os detalhes - mas é uma obra de referência e você provavelmente necessitará de uma introdução (a Borland tem igualmente disponível um manual Turbo Tutor).

Em resumo, a Borland deixou de lado, com o Trubo Pascal, algumas capacidades de outros compiladores, como a elevada memória (para códigos maiores que 64 Kb) e a capacidade de ligação a um código máquina pré-compilado durante a compilação. Em troca, produziram um sistema rápido, fácil de desenvolver e adequado à maior parte das necessidades de programação em computadores pessoais.

PECAN (UCSD) PASCAL

Este produto é fora do comum porque não se trata apenas de um compilador mas de um sistema operativo completo. O Pecan Pascal corre sobre o Power System, um ambiente operativo sobre o MS-DOS, e apresenta um editor, sistema de gestão de disco, utilitários para criação de ficheiros, copiador e, finalmente uma gama de compiladores.

O 'P-system' já foi desenvolvido há bastantes anos, e é uma tentativa de criar um sistema operativo standard que corra em diferentes máquinas, Para alcançar esse objectivo, o sistema compila programas para um determinado código máquina standard ('p-code'), que é suportado em cada máquina por um interpretador 'run-time'.

Por um lado, significa isto que se pode escrever programas que correrão em qualquer máquina desde um DEC a um Macintosh. Por outro lado, significa que não se podem correr os programas a partir do PROMPT A>, não podem ser utilizados sem o p-system (o qual requere uma licença para cada instalação) e que se tem que aprender o Power System.

LINGUAGEM

O Pecan Pascal segue o standard UCSD (University of California at San Diego), bastante completo e com ênfase especial no desenvolvimento de grandes projectos, a partir de unidades de compilação separadas. A linguagem é mais próxima do Pascal 'standard' do que do Turbo Pascal, mas as semelhanças são quase totais.

As novas capacidades incluem segmentação de procedimentos, e **'manipulação estrutural intrínseca'**. Existe igualmente suporte de processamento simultâneo - um programa que pode correr vários procedimentos ao mesmo tempo, com o P-system gerindo o tempo para cada um, enquanto correm. Também não existe limite de 64 Kb de código - o código máximo é de 16 Megabytes! Esta capacidade será pou-

co aproveitada no limite de 640 Kb do MS-DOS, mas bastante mais útil em outros computadores.

'POWER-FULL'

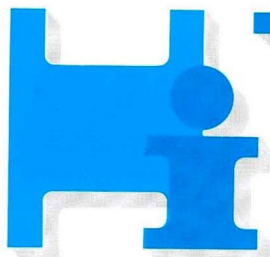
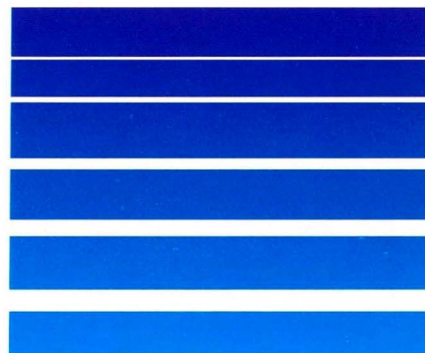
Apesar de todas estas capacidades, a linguagem em si não é mais difícil de escrever ou aprender que qualquer outro Pascal - mas, oh céus, aquele Power System!

É um ambiente operativo completamente diferente do MS-DOS, e parece que foi escrito noutro planeta qualquer. Como é que você percebe que #4:=, UTILS:$ é o comando para fazer o back-up dos ficheiros a partir do master disk? É obviamente muito poderoso, mas parece um daqueles obscuros sistemas operativos de alguns minicomputadores lá para 1975, - que foram provavelmente a sua fonte de inspiração.

O sistema utiliza 'volumes virtuais', que parecem ficheiros comuns ao MS-DOS, mas que estão divididos internamente em directorias e ficheiros, com o seu conjunto de comandos próprios. Inclusivé as disquetes podem ser formatadas de uma forma especial, que as torna ilegíveis em MS-DOS mas bastante mais eficientes sob P-system.

Pode aceder-se aos ficheiros MS-DOS a partir de programas Pascal, mas os programas necessitam estar em 'volumes virtuais'. O editor pode suportar ficheiros de qualquer tamanho mas, tal como o resto do sistema, é bastante complicado de usar - cheio de pequenos comandos em código que é necessário conhecer antecipadamente, faltando a informação de ajuda, e totalmente estranho aos habituais utilizadores de MS-DOS.

O Power System, incluindo o Pecan Pascal, é distribuído em três disquetes não protegidas, duas delas ilegíveis pelo MS-DOS. Para uma software house que queira produzir sistemas aplicacionais para uma grande variedade de computadores, é provavelmente um achado, este programa - trata-se de um Pascal bastante completo, existe uma variada gama de outras linguagens que podem ser facilmente integradas com aquela e, mais importante que tudo, o P-System torna os programas transportáveis para quaisquer outros computadores.

Para o fiel utilizador de MS-DOS, é sem dúvida menos atractivo - terá que aprender um sistema operativo inteiramente novo, não se produzem programas capazes de gravar e correr de imediato a partir das disquetes, e muitos pontos são realmente difíceis.

O HEXADECIMAL E O CÓDIGO MÁQUINA

Para quem tem um PC, o Código Máquina (assembler) é normalmente qualquer coisa de aterrador. Como contributos para a biblioteca do leitor, eis algumas subrotinas que poderão resolver alguns problemas dos programadores de assembler.

ASSEMBLAR 'À BORLA'

Em primeiro lugar seria interessante demonstrar como dispor de um programa de assembler sem ter de o comprar, roubar ou pedir emprestado. O MS-DOS que está no disco dos computadores pessoais, comporta algo que lhe permite começar a assemblar gratuitamente.

Em primeiro lugar, há que colocar a disquete do MS-DOS no computador pessoal, e digitar DEBUG. Mais ou menos um segundo depois aparecerá um sinal - o símbolo prompt do programa Debug.

De seguida digite o que vem a seguir, carregando na tecla "Return" após cada linha. Ignore por agora os números que vão aparecendo, já que serão explicados mais tarde:

```
a
mov dx,200
mov ah,9
int 21
ret
(prima apenas RETURN nesta linha)
a 200
db "Isto e uma mensagem $"
```

Entretanto, volte a carregar em Return e voltará ao 'prompt'. Deverá aparecer algo semelhante ao que se mostra na listagem da cópia do ecrã seguinte (não se preocupe se os primeiro quatro caracteres respondidos pela máquina não forem iguais aos que aqui são impressos):

```
-a
1A60:0100 mov dx,200
1A60:0103 mov ah,9
1A60:0105 int 21
1A60:0107 ret
1A60:0108
-a 200
1A60:0200 db "Isto e uma mensagem $"
1A60:0213
```

Se tudo estiver a correr bem, o utilizador deverá saber levar o computador a imprimir a mensagem entre aspas: o sinal de dólar é um delimitador.

Para ver se tudo está bem, carregue em G (de "Go", avançar). E eis o seu primeiro programa em Assembler, linguagem em relação à qual há sempre um certo ambiente de mistério, cultivado por aqueles que pretendem guardar a sabedoria para si próprios e manter o utilizador no estado de respeituosa ignorância.

Claro que se poderão experimentar outras mensagens escrevendo **a 200** outra vez, mais uma versão modificada da linha começada por **db**. Não devemos esquecer de concluir a mensagem com um sinal de dólar.

A sessão conclui-se introduzindo **q** para sair do Debug e voltar ao Sistema Operativo.

Vejamos agora o que é que acontecerá se chamarmos novamente o **Debug.** As linhas de assembler que foram introduzidas deverão estar ainda em memória, o que pode ser comprovado introduzindo o comando **u**, que lhe desassemblará o código ante os seus olhos, imprimindo-lhe no ecrã algo como isto:

(Seguido de um conjunto de linhas com o que estiver nesse ponto da memória).

```
1A60:0100 BA0002        MOV     DX,0200
1A60:0103 B409          MOV     AH,09
1A60:0105 CD21          INT     21
1A60:0107 C3            RET
```

Tente-se um outro Debug através do comando **r**, que produzirá o seguinte:

```
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1A60 ES=1A60 SS=1A60 CS=1A60 IP=0100   NV UP EI PL NZ NA PO NC
1A60:0100 BA0002        MOV     DX,0200
```

Não procure tentar perceber tudo isto à primeira. Antes de ir muito mais longe, vamos trilhar melhor os caminhos do sistema hexadecimal. Antes disso, vejamos as informações que nos foram apresentadas pelo Debug.

O assembler do 8086 tem um conjunto de "variáveis" de 16 bits chamadas registos, alguns dos quais de aplicação geral, enquanto outros desempenham papéis bem determinados.

Os primeiros quatro registos são AX,BX,CX e DX, que são os registos gerais de 16 bits.

Se olhar para a listagem que foi apresentada atrás quando se fez o Debug, repare que a mensagem estava colocada em 200- de facto, no endereço referenciado pelo computador como 1A60:0200. A explicação exacta para isto pode esperar.

Para poder imprimir a mensagem, a primeira coisa que o assembler precisa de saber é, muito naturalmente, onde é que esta se encontra.

Deste modo, a primeira instrução (a qual se encontra à espera de ser executada) é MOV DX.200 - por outras palavras, movimenta o endereço 200 para o registo DX.

Se digitar o comando **t** (trace), o Debug executará essa instrução ('single-stepping' é a palavra jargão para esse procedimento) podendo observar-se então o seguinte (repare-se nos novos elementos do registo DX):

Aparentemente,o que aconteceu foi que o 09 se dirigiu para a zona esquerda do registo AX. Isso acontece porque cada um dos quatro registos gerais AX, BX,CX e DX podem também ser encarados como estando divididos em dois registos de 8 bits.

A principal razão para isto é que a maior parte das vezes só precisamos de trabalhar com quantidades de 8 bit.

Os nomes dos registos (que reportam aos 'tempos antigos' do assembler do 8088) são:

```
AH            BH            CH            DH
+   =  AX     +   = BX      +    = CX     +     + DX
AL            BL            CL            DL
```

Não se preocupe se tudo isto se torna bastante confuso. Seria um grande feito perceber tudo à primeira vez e vai ser necessário mais algumas leituras para você perceber o que se passa.

Vamos agora ver por alto o que significa o pequeno programa que acabámos de escrever:

VER O QUE É MAIS IMPORTANTE

A primeira linha carrega o endereço do conjunto de caracteres que vão ser impressos no ecrã no registo DX.

```
AX=0000 BX=0000 CX=0000 DX=0200 SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1A60 ES=1A60 SS=1A60 CS=1A60 IP=0103   NV UP EI PL NZ NA PO NC
1A60:0103 B409          MOV     AH,09
```

Agora o DX contém o valor 200 e o computador está pronto para executar a instrução seguinte. Verá que se trata de um outro MOV (instrução de 'move') mas você não encontrará o registo AH no seu ecrã. Vejamos primeiro o que é que acontece quando digitamos **t:**

A segunda instrução (MOV AH,9) movimenta-se para o registo de 8 bit AH (que é a metade esquerda do registo AX) o que se considera ser uma chamada de função ao sistema operativo.

Existe, aliás, um grande número destas, e iremos encontrar outras mais à medida que formos entrando no 8086.

Esta chamada ao sistema operativo significa que a cadeia do registo DX está para ser impressa no ecrã (o sinal de dólar é o delimitador que dá a indicação ao sistema operativo onde é que se encontra o final da cadeia).

Isto é o que acontece quando lançamos o que se chama de interrupção (INT 21) no sistema.

Finalmente, o programa devolve (RET) o controlo para o Debug. De cada vez que o programa lança um INT 21 o sistema operativo "olha" para o registo de 8 bit AH para verificar que número está lá e age de acordo.

Contudo, convirá salientar, antes de mais, duas coisas: primeiro, que nesta etape é mais seguro sair do Debug e recarregá-lo de cada vez que se pretender escrever um novo programa.

E em segundo lugar, não se deve usar o comando **t** nas linhas que contenham INT 21.

Para ultrapassar o INT 21 descubra o endereço da instrução a seguir (no nosso exemplo é 107, o número a seguir aos dois pontos) e digite **g 107**, o que irá provocar a execução do programa até à instrução localizada em 107.

Neste ponto você já deve ter apanhado o essencial do assembler 8086. Será agora necessário, como foi dito mais atrás, deitar uma olhada ao hexadecimal.

Porquê utilizar o hexadecimal em computadores quando as pessoas normais trabalham em base 10 (o bom e velho decimal)?

Ora bem, se pensarmos um pouco chegamos à conclusão que não há nada de estranho no que se relaciona com diferentes bases numéricas: a semana tem 7 dias e não 10 (a CEE ainda não converteu o tempo para o sistema decimal), o minuto tem 60 segundos e o kilo tem 1000 gramas.

O HEXADECIMAL

A existência de base hexadecimal nos computadores e não da decimal, não se deveu a qualquer convenção de um qualquer organismo. A razão, como de sabe, reside no facto de os computadores se basearem na base 2 (binário) e seria terrivelmente entedioso ter de escrever sempre em binário, por exemplo, 11011111 por 223 em sistema

```
AX=0900 BX=0000 CX=0000 DX=0200 SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1A60 ES=1A60 SS=1A60 CS=1A60 IP=0105   NV UP EI PL NZ NZ PO NC
```

decimal.

Procurou tornar-se tudo muito mais fácil se, por exemplo, cada quatro bits (digitos binários) pudessem ser representados por um único dígito. E com o hexadecimal você pode fazer precisamente isso: deste modo, por exemplo 233 decimail é DF hexadecimal (o que convencinalmente referimos como ODFH, o H representando o hexadecimal) o qual poderia ser analisado em binário como 1101 + 1111.

O hexadecimal precisa de mais que os digitos 1 a 9, utilizando portanto ainda as letras de A a F. Se ainda se encontra confundido veja isto: 9 em decimal é 9H em hexadecimal, 10 é 0AH, 14 é OEH e 16 é 10H.

Neste ponto você bem pode estar de acordo com o cínico que uma vez disse: Se Deus tivesse querido que os humanos utilizassem o binário ter-nos-ia dado apenas dois dedos. Mas, no fundo, tudo bate certo.

Mas quais são os verdadeiros benefícios do hexadecimal? Vejamos um exemplo muito simples:

O 8086 utiliza o código ASCII (American Standard Code for Information Interchange) para representar caracteres. Existem, neste sentido, tabelas com os códigos dos caracteres, onde, por exemplo, o **Y** é representado pelo número 89 e o **y** (em minúsculas) pelo 121.

Traduza-o, contudo, para hexadecimal e algo de interessante acontecerá (a propósito, pode fazê-lo dividindo 89 por 16 e adicionando o resto como um digito entre 0-F): o **Y** é 59H e o **y** torna-se em 79H.

Em termos de hexadecimal a comparação funciona da seguinte maneira:

```
Y  =  1011001  =  59H
y  =  1111001  =  79H
```

O padrão é o mesmo, excepto para o segundo 1 na coluna do **y**.

Vamos agora explorar este facto numa situação que todos os criadores de programas encontram com uma regularidade quase monótona.

Se perguntar a um utilizador :" Quer cor?" (digite Y ou N). A forma mais difícil de encontrar o que o utilizador introduziu pelo teclado é pesquisar para Y e y e N, e de seguida para outros caracteres.

Se fosse possível reduzir todas as respostas para as maiúsculas então tudo seria mais simples. E tal pode ser feito.

Em primeiro lugar torna-se necessário introduzir um novo conceito, que utilizamos todos os dias sem nos darmos conta. Todos os dias dizemos para nós próprios(enfim, quase todos): "Se está a chover e a minha mulher levou o carro, se o meu filho mais velho tem o único guarda-chuva e o gato comeu a minha gabardina, então vou ficar todo molhado".

A LÓGICA BOOLEANA

Esta algebra do VERDADEIRO e do FALSO, chamada também de Booleana (pelo matemático George Boole), invadiu muitas áreas do nosso pensamento, pelo que não constitui qualquer surpresa que os computadores também a usem constantemente.

A função que vamos utilizar com as nossas letras maiúsculas e minúsculas é AND, que compara padrões de dois bit e devolve um zero se ambos não forem 'uns'.

Para conseguir AND utiliza-se uma mask (máscara), a qual, neste caso, é um número que já foi referido: ODFH. Se digitar AND 0DFH com, respectivamente, 79H e 59H, eis o que acontece:

```
79H   1111001  AND     59H   1011001  AND
0DFH  1011111   =      0DFH  1011111   =
      -------                -------
      1011001                1011001
```

Perdemos o bit que nos molestava em 79H e ambos saíram como 59H. Muito inteligente, não acha?

Vejamos agora o que está a correr sob o Debug. Carregue o Debug e use **a** para digitar o seguinte. De notar que o Assembler assume que os números estão em hexadecimal, portanto o h é desnecessário:

```
mov al,65
mov dl,df
and al,dl
mov dl,al
mov ah,2
int 21
ret
```

Utilize o **t** para seguir o programa até à instrução INT 21, e verá que o que se passa é que o 65 é carregado para o registo AL; a seguir o valor 0DFH é carregado para o DL.

As duas são 'ANDed' (conjunção) e o resultado aparece no AL. (65 torna-se ,por magia em 45).

O 45 é então copiado para o registo DL, onde o sistema operativo precisa de o encontrar se for imprimir no ecrã um E maiúsculo, utilizando a função call 2 no AH e lançando a interrupção, como é habitual.

Tente mudar o 65 e re-carregar o programa. Como é que se faz isto?

Saia do Debug utilizando **q** e imediatamente digite Debug. Verifique se o programa ainda lá está, utilizando o comando **u**.

Vejamos então um novo comando que necessita de um certo cuidado; o **e** permite-lhe examinar e, se o desejar, alterar os conteúdos da memória do seu PC.

O endereço que queremos mudar é 100 (estritamente 1A60:0100); digite então **e 100**, e verá isto no ecrã:

```
1A60:0100 B0.
```

Se suspeitou que o 0BH é o equivalente em código máquina à instrução MOV, então está cheio de razão. Como não nos interessa alterar isto, pressione a tecla de espaços.O resultado é:

```
1A60:0100 B0.    65.
```

O 65 não precisa de alteração. Vamos tentar mudar de e para f. Digite 66, e de seguida, prima Return. Se agora digitar **u** 100 verá a alteração e agora pode correr o programa para obter o resultado desejado.

Tente outras minúsculas (a gama em hexadecimal vai de 65H a 7AH, ou de 'a' a 'z'), não se esquecendo de sair e recarregar o Debug de cada vez.

Seria certamente muito maçador se fosse necessário escrever o seu programa em assembler de cada vez que se pretendesse corrê-lo. Existe uma forma de libertar o código executável do Debug, mas isso necessita de um certo cuidado.

Comece a partir do sistema operativo e digite **COPY CON FRED** (seguido de Return). Entre com uma linha de texto qualquer (o sistema não grava um ficheiro vazio), seguido de Control+Z e Return.

Você deve então ser informado de

que foi 1 file(s) foi copiado, e é o MS-DOS que o tem. Existe assim um ficheiro fictício no qual pode introduzir o seu programa. Para confirmar, pode utilizar-se o DIR e verificar se ele está lá.

Carregue o Debug, e torne a tarefa mais interessante, escrevendo um programa que peça a escrita de caracteres minúsculos, e as maiúsculas equivalentes serão impressas até pressionar a tecla de Return.

Utilizando **a**, introduza:

```
mov dx,200
mov ah,9
int 21
mov ah,1
int 21
mov dl,al
mov dl,df
and al,dl
mov dl,al
mov ah,2
int 21
cmp al,d
jnz 100
ret
```

Então, utilizando **a 200**, introduza:

```
db a
db d
db "Introduza uma letra em minusculas: $"
```

Para salvaguardar este programa, terá que executar os seguintes truques:

• Utlizando o comando **n** para dar nome ao ficheiro que pretende utilizar, digite **n fred**.

• O Debug precisa de ter nos registos BX e CX o número de bytes a serem escritos no ficheiro.

Vamos escolher 300, para termos uma boa margem de manobra. Digite **r bx** e terá impresso no ecrã o conteúdo do registo BX (provavelmente 0 nesta etapa) e ainda os dois pontos. Digite **300**, e depois Return.

Para saber se tudo isto está a funcionar bem, digite **r** para inspeccionar o conteúdo do registos. Repita o procedimento com o registo CX e verifique se tanto o BX como o CX contêm 300.

• Para escrever isto para o ficheiro, digite **w** e o Debug dir-lhe-á que está a escrever 300 bytes (300 em hexadecimal, isto é, 768 em decimal).

• Saia do Debug, e então digite **debug fred**, e terá então carregado o seu programa. Faça-o correr com **g**.

O que é que o programa faz? Imprime a mensagem, precedida de 0AF (line feed) e 0DF (carriage return). Pergunta de seguida por um caracter do teclado (trata-se da função de chamada, MOV AH,1).

De seguida, o 'ANDing' (conjunção) acontece; as maiúsculas equivalentes são impressas; e se não tiver premido Return, o programa salta para a endereço100 (JMP 100) e começa de novo.

O "line feed" e o "carriage return" no início da mensagem asseguram que a mensagem foi reescrita numa nova linha.

Não se esqueça que é necessário muito tempo até que consiga assimilar toda esta nova informação que lhe foi dada. Uma das maneiras de acelerar este processo de aprendizagem é a velha e eficaz técnica de praticar com o material que aqui foi exposto. Não se considere satisfeito até conseguir bons resultados e um certo à vontade neste campo. Boa sorte.

Exercícios

1.Como imprimiria duas mensagens, uma localizada no endereço 200, a outra no 300?

2.Sabendo que MOV AH,2 é a função de chamada para imprimir no ecrã um único caracter a partir do registo de 8 bit DL, e sabendo que o número hexadecimal 65 = 'e' minúsculo, como iria imprimir no ecrã a letra 'e'?

Respostas

1.Carregue o Debug e use o **a** para digitar:

```
mov dx,200
mov ah,9
int 21
mov dx,300
mov ah,9
int 21
ret
```

De seguida, utilizando „um 200" e um „300" introduza duas mensagens utilizando o mesmo formato do primeiro programa deste artigo.

2.Carregue o Debug e use o **a** para digitar:

```
mov dl,65
mov ah,2
int 21
ret
```

DISCO RAM

COMO provavelmente será do conhecimento dos nossos leitores, o AMSTRAD CPC 6128 possui 128Kb de RAM, dos quais 64K são utilizados da mesma maneira que os modelos 664 e 464. Quanto aos outros 64K , têm que ser manipulados através de comandos residentes. Estes encontram-se no disco do CP/M e denominam-se BANKMAN.

Ora bem, se num 464 ou 664 colocarmos uma ampliação de memória de 64K, também poderão utilizar o programa que vamos apresentar de seguida, já que ficam em igualdade de condições com o CPC 6128. Os leitores que tenham a paciência suficiente para digitar a listagem 1 e revê-la até que já não hajam erros, passarão a dispôr de um novo grupo de comandos residentes (precedidos sempre pela barra vertical "|") que lhes permitirá utilizar essa memória extra como se de um disco se tratasse, podendo manipular ficheiros como SAVE, LOAD, MERGE, CHAIN MERGE, OPENIN, OPENOUT, etc, etc. Os novos comandos são:

|DISCORAM: activa o Disco Ram, tanto para ficheiros de entrada como de saída.

|DISCORAM.IN: activa o Disco Ram só para ficheiros de entrada.

|DISCORAM.OUT: activa o Disco Ram só para ficheiros de saída.

|MAGNE: activa a cassete (fita magnética), tanto para ficheiros de entrada como de saída.

|MAGNE.IN activa a cassete só para ficheiros de entrada.

|MAGNE.OUT: activa a cassete só para ficheiros de saída.

|APAGA,<nome>: apaga do Disco Ram o ficheiro especificado em <nome>.

Os possuidores de um AMSTRAD CPC com unidade de disco podem utilizar indistintamente |TAPE ou |MAGNE, |TAPE.IN ou |MAGNE.IN e ainda |TAPE.OUT e |MAGNE.OUT.

Estes comandos de manipulação da fita magnética foram incluidos para os utilizadores de um 464 sem unidade de disco.

Os nomes para os ficheiros do Disco Ram podem ter até 10 caracteres de comprimento e todos os caracteres ASCII são válidos, não se utilizando extensão (embora nada nos impeça de chamar a um ficheiro PROG.BAS, uma vez que se admite que o ponto no meio do nome faça parte dele). O número de entradas da directoria está limitado a 20.

Quando o Disco Ram está activado para ficheiros de entrada (|DISCORAM ou |DISCORAM.IN), o comando CAT proporciona-nos um "dir" do Disco Ram. Neste "dir" aparece o nome de cada ficheiro com um símbolo que, à semelhança do que se passa quando trabalhamos com a cassete, indica-nos o tipo de programas armazenados:

$ BASIC desprotegido.
% BASIC protegido.
& BINARIO desprotegido.
' BINARIO protegido.
. ASCII

e, claro está, o tamanho do ficheiro em Kilobytes.

Uma vez executado, se as linhas de DATA estão correctas, o programa pergunta-nos se queremos armazenar o código, o que se torna bastante conveniente. Se utilizarmos o disco

```
10 ' *================================*
20 ' *           DISCO RAM            *
30 ' *           ---------            *
40 ' *       AMSTRAD CPC 128K         *
50 ' *       AMSTRAD MAGAZINE         *
60 ' *           (C) 1988             *
70 ' *================================*
80 '
90 DIREC=&A000:L=1000
100 MEMORY DIREC-1:RESTORE
110 READ A$
120 C=0:SOM=0
130 WHILE A$<>"FIM"
140 POKE DIREC,VAL("&"+A$)
150 SOM=SOM+VAL("&"+A$)
160 C=C+1:DIREC=DIREC+1
170 IF C<8 THEN 240
180 READ A
190 IF SOM=A THEN 220
200 PRINT"ERRO NA LINHA";L
210 END
220 C=0:L=L+10
230 SOM=0
240 READ A$
250 WEND
260 READ A
270 IF SOM=A THEN 300
280 PRINT"ERRO NA LINHA";L
290 END
300 PRINT"DADOS CORRECTOS"
310 ON ERROR GOTO 420
320 |TAPE
330 PRINT
340 INPUT"GRAVAR O CODIGO (S/N)";K$
350 IF UPPER$(LEFT$(K$,1))<>"S" THEN 390
360 INPUT"DISCO (S/N)";K$
370 IF UPPER$(LEFT$(K$,1))="S" THEN |DISC
380 SAVE"DISCORAM.BIN",B,&A000,&61F
390 CALL &A000
400 PRINT CHR$(7)"DISCO RAM INSTALADO"
410 END
420 IF ERR=28 AND ERL=320 THEN RESUME 390
430 PRINT CHR$(7)"ERRO"ERR"NA LINHA"ERL:END
1000 DATA C3,FB,A0,F5,C5,3E,C0,01,1303
1010 DATA 00,7F,ED,79,C1,F1,C9,F5,1365
1020 DATA C5,F6,C4,01,00,7F,ED,79,1125
1030 DATA C1,F1,C9,F5,C5,47,3E,20,1242
1040 DATA CD,5A,BB,78,06,00,FE,0A,872
1050 DATA 38,05,04,D6,0A,18,F7,4F,639
1060 DATA 78,FE,00,20,04,3E,20,18,528
1070 DATA 02,C6,30,CD,5A,BB,79,C6,1049
1080 DATA 30,CD,5A,BB,C1,F1,C9,E5,1394
1090 DATA D5,11,39,A6,48,06,0A,3E,603
1100 DATA 20,12,13,10,FC,41,11,39,476
1110 DATA A6,7E,12,23,13,10,FA,D1,839
1120 DATA E1,C9,78,B7,28,34,FE,0B,1086
1130 DATA 38,02,06,0A,CD,47,A0,D5,723
1140 DATA AF,CD,0F,A0,11,00,40,06,642
```

```
1150 DATA 14,C5,D5,21,39,A6,06,0A,702
1160 DATA 1A,BE,20,0B,13,23,10,F8,577
1170 DATA E1,C1,EB,D1,37,18,0C,D1,1162
1180 DATA 01,14,00,EB,09,EB,C1,10,709
1190 DATA E0,D1,AF,CD,03,A0,C9,7E,1303
1200 DATA FE,FF,C8,CD,5A,BB,23,18,1250
1210 DATA F6,E5,7C,E6,C0,CB,17,CB,1450
1220 DATA 17,CB,17,CD,0F,A0,7C,E6,983
1230 DATA 3F,F6,40,67,7E,CD,03,A0,970
1240 DATA E1,C9,E5,F5,7C,E6,C0,CB,1649
1250 DATA 17,CB,17,CB,17,CD,0F,A0,855
1260 DATA 7C,E6,3F,F6,40,67,F1,77,1190
1270 DATA CD,03,A0,E1,C9,F5,CD,03,1247
1280 DATA B9,32,33,A6,CD,09,B9,32,901
1290 DATA 34,A6,F1,C9,F5,3A,34,A6,1181
1300 DATA CD,0C,B9,3A,33,A6,CD,0C,894
1310 DATA B9,F1,C9,21,00,40,22,2E,804
1320 DATA A6,06,14,AF,CD,0F,A0,3E,809
1330 DATA FF,11,14,00,77,19,10,FC,704
1340 DATA CD,03,A0,AF,21,90,01,22,755
1350 DATA 1E,A6,32,31,A6,32,32,A6,727
1360 DATA 32,30,A6,CD,31,A5,3A,F6,987
1370 DATA A5,B7,C0,2F,32,F6,A5,01,1049
1380 DATA 3D,A5,21,43,A6,CD,D1,BC,1094
1390 DATA C9,3A,31,A6,B7,20,0D,CD,907
1400 DATA 62,A0,38,0B,21,8E,A5,CD,870
1410 DATA 9F,A0,AF,C9,AF,3C,C9,3E,1193
1420 DATA 03,32,31,A6,7C,E6,3F,67,788
1430 DATA CD,A9,A0,D5,DD,E1,DD,77,1533
1440 DATA 12,23,CD,A9,A0,DD,77,15,948
1450 DATA 23,CD,A9,A0,DD,77,16,23,966
1460 DATA CD,A9,A0,4F,DD,71,18,23,1006
1470 DATA CD,A9,A0,47,DD,70,19,23,998
1480 DATA CD,A9,A0,DD,77,1A,32,28,990
1490 DATA A6,23,CD,A9,A0,DD,77,1B,1102
1500 DATA 32,29,A6,ED,43,20,A6,C5,956
1510 DATA 23,CD,A9,A0,4F,23,CD,A9,1057
1520 DATA A0,47,ED,43,24,A6,EB,CD,1177
1530 DATA DD,A0,DD,5E,15,DD,56,16,1046
1540 DATA C1,F6,FF,DD,7E,12,37,CD,1319
1550 DATA EC,A0,C9,3A,31,A6,B7,20,1085
1560 DATA 06,3E,00,32,31,A6,C9,37,589
1570 DATA 18,F7,18,F5,E5,3A,31,A6,1042
1580 DATA B7,28,21,3E,01,32,31,A6,584
1590 DATA 2A,20,A6,7C,B5,28,15,2B,649
1600 DATA 22,20,A6,2A,24,A6,CD,A9,850
1610 DATA A0,23,22,24,A6,6F,F6,FF,1043
1620 DATA 7D,37,E1,C9,F6,FF,18,FA,1381
1630 DATA 3A,31,A6,FE,03,20,27,3E,663
1640 DATA 02,32,31,A6,ED,4B,20,A6,777
1650 DATA ED,5B,24,A6,78,B1,28,0B,878
1660 DATA 0B,EB,CD,A9,A0,23,EB,77,1169
1670 DATA 23,18,F1,ED,43,20,A6,2A,844
1680 DATA 28,A6,F6,FF,37,C9,F6,FF,1464
1690 DATA C9,E5,2A,24,A6,2B,22,24,787
1700 DATA A6,2A,20,A6,23,22,20,A6,673
1710 DATA E1,C9,3E,01,32,31,A6,E5,983
1720 DATA 2A,20,A6,7C,B5,28,AD,F6,1004
1730 DATA FF,37,E1,C9,3A,32,A6,B7,1193
1740 DATA C2,4C,A1,CD,62,A0,30,08,950
1750 DATA 21,9E,A5,CD,9F,A0,AF,C9,1256
1760 DATA 3E,03,32,32,A6,CD,77,A2,817
```

devemos gravá-lo no mesmo disco em que fique a listagem 2. Se utilizarmos a cassete, devemos gravá-lo mesmo antes da listagem 2. Deste modo, para usar o Disco Ram bastará executar a listagem 2, poupando assim o tempo da verificação das linhas de DATA e do "pokeado" destas.

Suponhamos que acabámos de criar um ficheiro no Disco Ram com o nome de JOGO-V1. Se o quisermos apagar deveremos digitar |APAGA,"JOGO-V1". É tão simples como isto. Convém referir que ||APAGA apaga sempre um ficheiro do Disco Ram, independemente de estar ou não seleccionado. Contudo, o Disco Ram não admite astúcias nos nomes como acontece no AMSDOS (ao falar de astúcias referimo-nos a "*" e ?"). Desta maneira não podemos apagar vários ficheiros de cada vez como |APAGA,"JO*". Se, contudo, quisermos apagar todos os ficheiros do Disco Ram podemos fazê-lo com CALL &A000.

As vantagens que nos proporciona o dispôr de um Disco Ram (embora só com uma capacidade de 64K) são, por um lado, a rapidez de acesso e, por outro, uma poupança de espaço e desgaste nos discos — no caso, claro, de querermos guardar os ficheiros criados.

Esperamos que este programa vos interesse e vos possa ser útil. Ânimo, portanto, e toca a bater no teclado.

```
1770 DATA EB,22,2A,A6,E5,DD,E1,DD,1373
1780 DATA 36,12,16,F6,FF,37,C9,3A,909
1790 DATA 30,A6,FE,14,28,25,AF,CD,945
1800 DATA 0F,A0,D5,01,0A,00,21,39,489
1810 DATA A6,ED,5B,2E,A6,ED,B0,3E,1181
1820 DATA 16,12,D1,2A,1E,A6,22,26,559
1830 DATA A6,21,00,00,22,22,A6,CD,638
1840 DATA 03,A0,C9,E1,21,B3,A5,CD,1171
1850 DATA 9F,A0,AF,CD,03,A0,C9,3A,1121
1860 DATA 32,A6,B7,20,08,F6,FF,3E,1002
1870 DATA 00,32,32,A6,C9,AF,CD,0F,862
1880 DATA A0,2A,22,A6,7C,B5,28,47,818
1890 DATA DD,2A,2E,A6,DD,75,0D,DD,1047
1900 DATA 74,0E,2A,1E,A6,DD,75,11,723
1910 DATA DD,74,12,DD,E5,CD,03,A0,1173
1920 DATA DD,2A,2A,A6,DD,46,12,AF,955
1930 DATA CD,0F,A0,78,DD,E1,DD,77,1286
1940 DATA 0A,2A,26,A6,22,1E,A6,3A,544
1950 DATA 30,A6,3C,32,30,A6,01,14,559
1960 DATA 00,DD,09,DD,22,2E,A6,CD,902
1970 DATA 03,A0,F6,FF,37,18,A8,AF,1086
1980 DATA CD,0F,A0,3A,30,A6,FE,14,926
1990 DATA 28,05,2A,2E,A6,36,FF,CD,813
2000 DATA 03,A0,18,E6,E5,F5,3A,32,999
2010 DATA A6,FE,01,28,09,FE,03,20,759
2020 DATA 20,3E,01,32,32,A6,2A,26,441
2030 DATA A6,7C,B5,28,19,F1,CD,C2,1176
2040 DATA A0,23,22,26,A6,2A,22,A6,675
2050 DATA 23,22,22,A6,E1,F6,FF,37,1050
2060 DATA C9,F1,E1,F6,FF,C9,21,C8,1602
2070 DATA A5,CD,9F,A0,F1,E1,AF,C9,1531
2080 DATA F5,3A,32,A6,FE,03,20,4D,885
2090 DATA 3E,02,32,32,A6,F1,DD,2A,834
2100 DATA 2A,A6,DD,77,12,F5,AF,CD,1191
2110 DATA 0F,A0,F1,DD,2A,2E,A6,DD,1112
2120 DATA 77,0A,DD,71,0F,DD,70,10,827
2130 DATA DD,75,0B,DD,74,0C,CD,03,906
2140 DATA A0,ED,53,22,A6,4B,42,EB,1056
2150 DATA 78,B1,28,15,2A,26,A6,7C,728
2160 DATA B5,28,16,EB,E7,EB,CD,C2,1343
2170 DATA A0,23,22,26,A6,13,0B,18,487
2180 DATA E7,F6,FF,37,C9,F1,F6,FF,1730
2190 DATA C9,21,C8,A5,CD,9F,A0,AF,1298
2200 DATA C9,21,DC,A5,CD,9F,A0,AF,1318
2210 DATA CD,0F,A0,DD,21,00,40,12,716
2220 DATA CD,DD,A0,DD,7E,00,FE,FF,1442
2230 DATA 28,3F,DD,E5,E1,06,0A,7E,920
2240 DATA CD,5A,BB,23,10,F9,3E,20,876
2250 DATA CD,5A,BB,7E,E6,0F,C6,24,1087
2260 DATA CD,5A,BB,DD,7E,0E,CB,3F,1109
2270 DATA CB,3F,3C,CD,1B,A0,3E,4B,855
2280 DATA CD,5A,BB,3E,0D,CD,5A,BB,1039
2290 DATA 3E,0A,CD,5A,BB,01,14,00,575
2300 DATA DD,09,1A,3C,12,FE,14,20,640
2310 DATA BA,3E,0D,CD,5A,BB,3E,0A,815
2320 DATA CD,5A,BB,F6,FF,37,CD,03,1246
2330 DATA A0,CD,EC,A0,C9,FE,01,C2,1411
2340 DATA 44,A1,DD,6E,00,DD,66,01,884
2350 DATA 46,23,5E,23,56,EB,CD,62,858
2360 DATA A0,D2,44,A1,AF,CD,0F,A0,1154
2370 DATA E5,DD,E1,01,0A,00,ED,42,989
2380 DATA DD,4E,03,DD,46,04,ED,43,901
2390 DATA 2C,A6,EB,2A,1E,A6,AF,ED,1095
```

```
2400 DATA 42,22,1E,A6,EB,36,FF,7D,965
2410 DATA FE,7C,28,5C,DD,7E,0A,FE,1121
2420 DATA FF,28,55,E5,DD,5E,07,DD,1152
2430 DATA 56,08,DD,6E,1B,DD,66,1C,803
2440 DATA ED,4B,1E,A6,CD,A9,A0,EB,1277
2450 DATA CD,C2,A0,EB,23,13,7A,B8,1154
2460 DATA 20,F2,7B,B9,20,EE,E1,54,1161
2470 DATA 5D,01,14,00,09,AF,CD,0F,518
2480 DATA A0,ED,B0,E5,DD,6E,07,DD,1361
2490 DATA 66,08,AF,ED,4B,2C,A6,ED,1044
2500 DATA 42,DD,75,07,DD,74,08,01,757
2510 DATA 14,00,DD,09,E1,3E,FF,12,810
2520 DATA 01,14,00,7B,FE,7C,20,D9,771
2530 DATA 3A,30,A6,3D,32,30,A6,2A,639
2540 DATA 2E,A6,01,14,00,AF,ED,42,711
2550 DATA 22,2E,A6,CD,03,A0,C9,CD,1020
2560 DATA E5,A4,C3,FC,A4,01,15,00,1026
2570 DATA 21,F7,A5,11,77,BC,ED,B0,1182
2580 DATA 01,03,00,21,0C,A6,11,9B,387
2590 DATA BC,ED,B0,C9,01,0F,00,21,851
2600 DATA 0F,A6,11,8C,BC,ED,B0,C9,1140
2610 DATA CD,0E,A5,C3,25,A5,01,15,803
2620 DATA 00,21,47,A6,11,77,BC,ED,831
2630 DATA B0,01,03,00,21,6B,A6,11,503
2640 DATA 9B,BC,ED,B0,C9,01,0F,00,973
2650 DATA 21,5C,A6,11,8C,BC,ED,B0,1049
2660 DATA C9,21,77,BC,11,47,A6,01,796
2670 DATA 27,00,ED,B0,C9,54,A5,C3,1097
2680 DATA DF,A4,C3,E5,A4,C3,FC,A4,1586
```

```
2690 DATA C3,08,A5,C3,0E,A5,C3,25,974
2700 DATA A5,C3,2D,A4,44,49,53,43,860
2710 DATA 4F,52,41,CD,44,49,53,43,722
2720 DATA 4F,52,41,4D,2E,49,CE,44,696
2730 DATA 49,53,43,4F,52,41,4D,2E,572
2740 DATA 4F,55,D4,4D,41,47,4E,C5,864
2750 DATA 4D,41,47,4E,45,2E,49,CE,685
2760 DATA 4D,41,47,4E,45,2E,4F,55,570
2770 DATA D4,41,50,41,47,C1,0D,0A,709
2780 DATA 43,6F,6D,61,6E,64,6F,80,833
2790 DATA 3F,21,21,0D,0A,FF,0D,0A,430
2800 DATA 4E,6F,6D,65,80,3E,80,65,818
2810 DATA 78,69,73,74,65,6E,74,65,884
2820 DATA 0D,0A,FF,0D,0A,44,69,72,588
2830 DATA 65,63,74,6F,72,69,6F,20,789
2840 DATA 6C,6C,65,6E,6F,0D,0A,FF,816
2850 DATA 0D,0A,44,69,73,63,6F,20,553
2860 DATA 52,61,6D,20,6C,6C,65,6E,747
2870 DATA 6F,0D,0A,FF,0D,0A,44,69,585
2880 DATA 72,65,63,74,6F,72,69,6F,871
2890 DATA 20,44,69,73,63,6F,20,52,644
2900 DATA 61,6D,3A,0D,0A,FF,00,C3,737
2910 DATA 39,A1,C3,BB,A1,C3,CA,A1,1319
2920 DATA C3,CC,A1,C3,F8,A1,C3,29,1400
2930 DATA A2,C3,3A,A2,C3,C1,A3,C3,1323
2940 DATA 4C,A2,C3,AF,A2,C3,0F,A3,1143
2950 DATA C3,24,A3,C3,60,A3,00,FIM,848
```

CORREIO DOS LEITORES

ASSINATURA SIM, MAS A PARTIR DE QUANDO?

Fernando Moreira da Costa — LOULÉ

AM: — O "problema" das assinaturas da AM parece ter afectado muito boa gente quando na realidade o que se passava nunca chegou a ser um problema. Os Leitores mais observadores provavelmente já constataram que entre os postais destinados á assinatura da revista publicados nos números 1 e 2, e os que mais recentemente temos vindo a publicar existem algumas diferenças. Com efeito, para além das diferenças evidentes devido a campanhas especiais, os postais publicados a partir do N.º 3 da AM passaram a incluir um espaço destinado a informação por parte do novo assinante, de qual o número da revista em que deseja iniciar a assinatura. Esta alteração surgiu basicamente como uma necessidade nossa para melhor poder satisfazer os leitores.
O que estava a acontecer até há pouco tempo era os leitores assinarem a revista a pensar que a assinatura incluiria o número X e, devido ao tempo que demora a organizar em termos burocráticos a sua inclusão no número de assinantes, esta apenas poder ser considerada a partir do número X+1. Esta situação levava os novos assinantes a pensarem que algo de anormal se passava em relação à sua assinatura quando apenas passavam por uma fase normal do processo pelo qual todos os outros já tinham passado. Muito simples dito desta maneira, tudo isto parece ter causado o "pânico" entre os leitores que, ao verem no mercado o número que pensavam receber já pelo correio, não sabiam se o facto de não o receberem se tratava de um atraso nosso, ou se deviam adquirir ainda esse número no local habitual.
Colocando um ponto final na questão, decidimos abrir o referido espaço no postal de assinatura com o fim de tornar claro qual o n.º da AM que é considerado como início de assinatura. É claro que o envio do primeiro número, ou números (já que qualquer pessoa poderá fazer uma assinatura a partir do número 1 da revista), pode sofrer um pequeno atraso de alguns dias em relação ao que seria desejável, embora façamos sempre todos os possíveis para o tentar minoria. Como referimos, e reafirmamos, esse atraso de meia dúzia de dias apenas se verifica, no entanto, no primeiro envio referente à assinatura da AM, todos os outros se têm processado e, vão continuar a processar-se, dentro dos períodos normais.
Com esta resposta pensamos ter respondido não só ao Fernando Moreira mas também a todos os outros leitores que nos escreveram pela mesma razão, já que, como é lógico, não teria qualquer sentido "encher" várias páginas da AM com cópias da mesma resposta a cartas de leitores diferentes.

NÚMEROS ATRASADOS: COMO CONSEGUI-LOS?

José Alberto Oliveira — BRAGA

AM: — Mais uma vez, uma preocupação comum a muitos leitores que nos começaram a acompanhar depois de já terem passado pelo mercado alguns números da AM. A obtenção desses mesmos primeiros números da revista. Assim, para esclarecer todos os leitores que, tal como o José Oliveira, desejem adquirir números atrasados da AM, podemos informar que esta aquisição poderá ser feita mediante envio de cheque ou vale postal no valor do preço de capa da revista (300$00 ou 350$00), adicionado ao valor dos portes, e emitido a favor de Publinfor, Publicações e Comérico de Artigos de Informática, S.A.. Seguido este procedimento, na volta do correio, apenas restará abrir o envelope com as revistas pedidas afim de completar a colecção.
A alternativa, para quem não possui ainda um único exemplar da AM, passa, por exemplo, pela assinatura a partir do número 1.

— Desde o início, e já lá vai algum tempo, pois adquiri o PCW logo que eles apareceram no mercado, que procurei saber da existência de uma publicação explicativa do BASIC Mallard. Cheguei a escrever para a Amstrad em Inglaterra que me endossou para a sua representante em França, dada a minha preferência pela língua francesa. Não consegui nada, mesmo com a interferência de alguém morador em Paris. Por isso renovo a minha pergunta: Há algum processo de satisfazer a minha curiosidade, uma vez que o manual é de todo insuficiente para o efeito?
— No segundo número são anunciados alguns programas, mas nem em todos é referido o modelo a que se aplicam, de modo que pergunto: o ABILITY corre no PCW? O SuperCalc 3.1 corre no PCW? E quanto a este último, mais pergunto, se a resposta for positiva: ganho alguma coisa em o adquirir, uma vez que possuo já o SuperCalc 2?
— Na pág.39 são indicadas várias chaves para programar em BASIC, de entre as quais eu saliento as que se referem aos tipos de letras, linhas 260 e 270, que não consegui fazer funcionar, isto é, obter qualquer alteração das letras. De resto, eu para obter outro tipo de letra, quer em BASIC quer em CP/M, faço sempre o seguinte: entro no BASIC e digito LPRINT CHR$(15). Depois tanto faz imprimir em BASIC como em CP/M que obtenho uma letra de tipo mais pequeno. Mas gostaria de saber porque não dá o vosso método. Terá alguma coisa a ver com o facto de o meu PCW ser originariamente 8256?

Rodolfo Soeiro de Sousa — Lisboa

AM: — Na realidade existem várias maneiras de satisfazer a sua curiosidade. De todas as possíveis, talvez as mais adequadas, e simples, sejam: entrar em contacto com a Locomotive Software, em Inglaterra, ou colocar-nos todas as suas dúvidas nesta secção, embora seja, sem dúvida, a mais interessante para si, já que na eventualidade de não existir um manual mais completo sobre o assunto em que possui dúvidas, poderá contactar com os programadores da softhouse que desenvolveu a linguagem, possui como único inconveniente a morosidade que lhe é inerente.
A segunda solução, que como referimos consistiria em colocar-nos os seus problemas e dúvidas, sendo muito mais rápida do que a que anteriormente referimos, possui talvez como único inconveniente o facto de ter de ser dirigida. Isto é, enquanto, se optar pela primeira solução, pode solicitar aos homens da Locomotive que lhe enviem tudo aquilo que possuem sobre o dialecto de BASIC que comercializam, se optar pela segunda solução nunca poderá obter mais do que a resposta, ainda que minimamente detalhada, a dúvidas expressas por escrito e de forma concreta. É evidente, que nunca poderemos explicar nesta secção todos os detalhes do BASIC Mallard sob a forma de lições envolvendo temas gerais. No "corpo" da revista, contudo, inseridos nas devidas secções, é muito provável que apareçam com alguma frequência artigos subordinados ao tema em causa.
— A observação que nos fez nesta pergunta, embora só tenha eco em termos de resposta neste número da AM, foi já a seu tempo, e em conjunto com algumas outras de diferentes leitores, mas idêntico teor, elemento importante para melhoramento da secção Clube AM, que, como pode comprovar, desde o n.º 3 da AM, inclui em cada artigo uma "etiqueta" indicadora do computador a que o mesmo se destina. Apesar disso, e uma vez que continua a existir a possibilidade de tal facto não ter respondido às suas dúvidas, podemos informá-lo de que ambas os "packages" referidos (ABILITY e SuperCalc 3.1) se destinam a computadores pessoais compatíveis IBM, e que, como tal, não "correm" nos computadores Amstrad das gamas CPC e PCW.
— Com efeito, as linhas que nos refere na secção TRUQUES PCW (AM N.º2), não alteram o tipo de letra impresso porque não é essa a sua finalidade. As referidas linhas, que como todas as outras "funcionam" correctamente, quando utilizadas apenas alteram a densidade de caracteres por linha. A linha 260, como aliás se refere no seu final, proporciona a impressão de texto com 12 caracteres por polegada, num tipo de impressão normalmente designado por ELITE, enquanto que a linha 270 permite a impressão em PICA (10 caracteres por polegada). Relembramos mais uma vez que nenhuma das operações referidas afecta o tipo ou corpo da letra, razão pela qual lhe pareceu que estas não "funcionavam" depois de activadas.
Como última observação deve notar-se que a alteração que fez ao seu PCW, apenas teve como consequência o aumento das capacidades da máquina sem alterar em tudo o resto as devidas performances. Dificilmente essa alteração lhe provocará problemas do tipo dos que questiona.

CORREIO DOS LEITORES

— Qual é o limite (aconselhável) de tempo que o Amstrad 1512 poderá estar ligado?
— Haverá algum inconveniente em deixar a ficha ligada com o computador desligado?
— Que produtos de limpeza para o computador me aconselham?
— Como obter os efeitos gráficos dos GEM (cursor em cima de um fundo com gráficos) num programa concluído em GW BASIC, sem que o CPU tenha de chamar ecrãs e tornar a apagá-los para que o "rasto" do cursor não se note?

Paulo Frederico Brito — Lisboa

AM: — Na realidade, nunca alguém se deu ao trabalho de ligar o PC 1512 e de se sentar em frente do ecrã à espera que ele "estoirasse". Digamos que, para além de ser uma tarefa pouco interessante e de requerer uma grande dose de paciência por parte do humano envolvido na experiência, este teste implicaria uma quantidade considerável de tempo livre para se poder colocar em prática, o que, actualmente, poucas pessoas se podem regozijar de possuir. A nossa experiência neste domínio apenas nos permite afirmar que pelo menos alguns destes computadores podem suportar mais de 3 000 horas de trabalho ininterrupto, período de tempo até hoje suportado por uma das máquinas que controla entradas e saídas numa empresa do Norte. É claro que, apesar de tal máquina desempenhar ainda neste momento (e espera-se que por longos meses) a tarefa referida sem um único minuto de descanso, não é impossível que depois de apenas uma hora de trabalho um PC 1512 possa "estoirar" por qualquer razão imprevisível. Não nos devemos esquecer que estamos a falar de máquinas, e que muitas vezes estas também nos pregam as suas "partidas".
Como última nota, só não queremos deixar de relembrar que o PC 1512 não tem tempos de funcionamento especiais, e que, embora não possamos garantir o MTBF como se se tratasse de uma cabeça de impressora, por exemplo, acreditamos que este computador possui a mesma fiabilidade de qualquer outro do mesmo tipo existente no mercado.
Em caso de trabalho prolongado não devemos, no entanto, esquecer as condições em que o fazemos, já que essas sim, poderão constituir causa de avaria.
— Em princípio diríamos que não existe inconveniente algum, Contudo, depois de analisarmos a questão lembramo-nos de um possível inconveniente. Se por acaso o fio que liga o PC à rede, se encontrar estendido no chão por forma a que possa tropeçar nele, talvez seja melhor desligar a máquina depois da sessão de trabalho diária.
Em resumo, e depois da brincadeira, resta-nos sublinhar que tecnicamente não vemos qualquer inconveniente em deixar a ficha ligada à rede com o PC desligado no interruptor.
— Sem referir, pelo menos por agora (já que o assunto poderá constituir matéria para um trabalho mais extenso noutra secção desta revista), marcas de produtos utilizáveis para a limpeza adequada do seu computador pessoal, podemos aconselhá-lo a limpá-lo exteriormente com um produto não abrasivo, e não do tipo "spray". Os "sprays" que, como se sabe, só por si já são inconvenientes por provocarem rupturas na camada de ozono da atmosfera, neste caso podem entrar pelas ranhuras de respiração-ventilação das diferentes unidades, provocando curto-circuitos no momento em que a energia eléctrica percorre as placas de circuito impresso.
O processo adequado de limpeza será, portanto, coseguido através de um pano humedecido num líquido não abrasivo, que se poderá passar pelas superfícies a limpar, sendo estas em seguida limpas de novo com um pano seco.
— Veja resposta no próximo número.

— Na revista Amstrad Magazine n.º2 li que os computadores Amstrad poderiam vir a ser vendidos por prestações. Gostaria de saber se é para breve ou não. Gostaria também, que me enviassem uma lista com os preços dos Amstrad PC 1512 e PC 1640, e dos PPC 512 e PPC 640.

João Fernando Ferreira — Ovar

AM: — Pecando pelo atraso a nossa resposta à primeira parte da sua pergunta encontra-se bastante explícita na secção NOTÍCIAS do número anterior da AM. Aí, como certamente pode observar, estão explicados os detalhes mais importantes do serviço CREDI-AMSTRAD, o que torna deslocada, e de certa forma inoportuna, a repetição da resposta neste espaço.
No que diz respeito ao segundo pedido que nos faz, lamentamos mais uma vez não poder satisfazê-lo visto não existir um preço de mercado fixo para a comercilaização deste tipo de equipamentos. Os representantes de todas as marcas de computadores pessoais limitam-se a aconselhar preços de venda ao público aos seus "dealers", deixando nas mãos destes últimos a sua fixação. Por este facto se justifica que possamos encontrar o mesmo tipo de computador em lojas diferentes, a preços diferentes. Por esta razão também, uma eventual lista por nós fornecida de acordo com os preços praticados por uma loja específica, pode não ter qualquer relação com os que a loja onde costuma adquirir o seu equipamento pratica, e não possui, por isso, qualquer interesse prático.

— Como adquirir o emulador CGA para Hércules?

Carlos Alberto Santana — Lisboa

AM: — No momento em que escrevemos esta resposta, conhecemos no mercado cerca de 4 ou 5 emuladores de CGA para a carta gráfica Hércules. Poderíamos sugerir-lhe vários modos possíveis para a aquisição de um deles, contudo, parece-nos que o mais simples, e económico, talvez seja a aquisição de uma das diskettes que colocamos à sua disposição na secção FREE SOFT, visto que, todas elas, para além de incluirem os utilitários/jogos referidos nesse espaço, incluem um emulador de CGA para Hércules, afim de permitir aos utilizadores nas suas condições, a utilização de todas as packages que utilizam essa carta gráfica.
O referido emulador poderá posteriormente ser utilizado por si para "correr" todos os programas (inclusivé, como é lógico, os jogos) que utilizam a CGA, sem quaisquer problemas ou dificulrdades daí resultantes.

— Possuo um amstrad CPC 464, e na localidade onde resido não consigo obter jogos didáticos ou programas científicos.
Desejava saber se num futuro próximo a AM poderia inserir no "CLUBE DOS LEITORES" a venda de jogos por encomenda?

José Carlos David — Ferreira do Zezere

AM: — Como pode constatar, mesmo através da referida secção nesta revista, a sugestão que extraímos da sua pergunta já começou a dar resultados. Desde o número 3 que começamos a proporcionar aos leitores a aquisição de alguns programas para CPC, e desde sempre pretendemos continuar a suportar os utilizadores destas máquinas, não só em termos de ajuda técnica e apresentação de trabalhos, quer no domínio do software, quer no domínio do hardware, como também ao nível da disponibilidade de "packages" comerciais.
Pensamos com estas afirmações ter respondido à sua questão, e ter contribuído para o manter confiante na máquina que utiliza, que como muitos utilizadores podem comprovar, continua a ser um excelente microcomputador.

— Na segunda edição, quando falavam do programa MIRROR II, referem-se várias vezes ao MODEM. Sei que este tem ligações com o telefone, mas podiam explicar-me mais detalhadamente de que se trata e qual o seu preço para o PC 1512?
— Indicavam-me por favor o preço da LQ 3500?

Nuno José Almeida Andrade — Coimbra

AM: — A palavra MODEM surge definida na maior parte dos dicionários de informática como um acrónimo para MODulator/DEModulator (modulador/"desmodulador"), e é aí indicada como a designação de um dispositivo que permite a transferência de DATA através dos circuitos telefónicos. Por outros termos, podemos dizer que o MODEM é um circuíto electrónico que permite a passagem de um sinal digital a analógico, e de um sinal analógico a digital.
Mas para que serve o modem? em apenas duas ou três palavras e baseado em ideias muito sim-

ples pode explicar-se a utilidade do modem se pensarmos nos elementos envolvidos no processo de transmissão de DATA anteriormente referido. Assim, como é do conhecimento comum, as linhas telefónicas normais têm como função principal o transporte de som no espaço, restringindo-se este som, na maior parte dos casos à voz humana. Logo, o tipo de circuitos utilizados em ambos os terminais de um sistema telefónico (ou linha telefónica) estão adequados à recepção e trasmissão de sinais analógicos, codificando-os e descodificando-os sempre que há uma emissão ou recepção de informação. Por este motivo, para um computador conseguir passar informação através da linha telefónica, deve utilizar-se como interface um circuito que lhe permita converter todos os dados em sinais analógicos, e vice-versa, afim de que o computador e o "terminal" telefónico se entendam durante o diálogo que estabelecerem.
Dentro dos diversos modem's existentes no nosso país, procurámos saber o preço dos que são representados pela COMINFOR (representante Amstrad em Portugal) e verificámos que esta entidade colocou no mercado dois dispositivos deste tipo (um com ligação via RS 232C, outro ligado num dos slots do PC) com um custo que se situa entre os 32 os 42 mil escudos acrescido do Imposto do Valor Acrescentado.
— A LQ 3500 é comercializada a um preço que na maior parte dos casos se situa entre os 85 e os 90 mil escudos + IVA. Como já temos referido, a precisão em termos de preços de mercado é sempre difícil (senão mesmo impossível), já que raramente existe um P.V.P. (preço de venda ao público) fixo. Normalmente os representantes dos diversos produtos apenas aconselham os P.V.P.'s.

— Quais as possibilidades de ligar o meu PCW 8256 à rede VIDEOTEX?
— Quais os periféricos e acessórios de que necessito e por quem são comercializados?

António Lourenço Pereira — Miranda do Corvo

AM: — A resposta à sua primeira questão pode resumir-se numa única palavra: Todas.
— A segunda questão deve no entanto ser objecto de uma resposta mais "trabalhada". Assim, para conseguir ligar o seu PCW a rede VIDEOTEX apenas necessita de um modem e de software de comunicações adequado à sua máquina. Esta necessidade embora pareça constituir um grande problema, na verdade limita-se a ser um pequeno "semi-problema". Passamos, portanto, a explicar o que pretendemos dizer com isto. Supondo que queremos mesmo ligar o PCW à rede VIDEOTEX, e que esta não foi apenas uma situação hipotética com o fim de esclarecer algumas dúvidas, podemos começar por adquirir o modem utilizável pelo PCW (SOFT 6064), comercializado pelo representante da Amstrad no nosso país, procurar obter em Inglaterra software de comunicações adequado para esta máquina, uma vez que neste lusíado canto ele não existe. Esta última fase do processo pode ser concluída com alguma facilidade através do serviço IMPORT que já desde há alguns números criámos para si. Uma vez na posse de ambos os "acessórios" é tudo uma questão de ler os manuais e... cuidado com os recibos de telefone ao fim mês.
Nota: se se interessa por estes assuntos não deixe de ler a AM. Num dos próximos números terá explicações e, sobre tudo o que diz respeito aos processos de comunicação de data.

— Será que o Bingo funciona com o CPC 464?

António Lopes Barreto — Portalegre

AM: — Infelizmente o programa "BINGO" publicado no n.º1 da AM não funciona no CPC 464, no entanto, o 464 está longe de ser esquecido por nós, e prova deste facto são os programas (jogos e técnicos) que comercializamos no CLUBE AM, e os diversos programas e rotinas que temos publicado e continuaremos a publicar para esta máquina.
Como já pode constatar neste momento, a sua sugestão de publicarmos alguns programas para o 464 tem sido, e será sempre, considerada e colocada em prática, uma vez que não é só em França (como nos refere na carta) que o CPC 464 é conhecido. Ao longo dos vários números da AM, uns falarão mais do 464 do que outros mas, como compreende, por diversas razões não podemos dedicar sempre o espaço que todos desejaríamos a cada tipo de computador Amstrad. Para já fique com a certeza de que terá muito mais programas para teclar nos próximos tempos como consequência de ler estas páginas.

— Há alguns meses, adquiri um computador PHILIPS VG 8020 MSX BASIC, por precisar de adquirir alguns conhecimentos de informática, para aplicação nos estudos de arquitectura. Assim, venho pedir-vos o favor de me informarem se todos os programas apresentados na vossa revista são compatíveis com o meu sistema? Se não o que devo modificar?

Jorge Manuel Rocha — Tomar

AM: — Os programas publicados na AM destinam-se, ou aos computadores pessoais, que na sua classe conseguiram atingir uma razoável percentagem de compatibilidade graças à IBM, ou aos diversos modelos de micros Amstrad que só são compatíveis entre eles. Simplificando a resposta à sua questão diremos, portanto, que todos os programas que surgirem publicados na AM os que mais facilmente se adaptam ao seu computador serão os que forem concebidos em GW-BASIC, ou BASICA (ambos dialectos do BASIC Microsoft), visto serem apresentados numa linguagem próxima da que o seu micro utiliza (tanto o GW-BASIC como o BASICA, e o MSX-BASIC são dialectos de uma mesma linguagem — o BASIC Microsoft).
Apesar da adaptação de programas entre as linguagens referidas se poder efectuar, na maior parte dos casos, com alguma facilidade, ela requer alguns conhecimentos por parte do utilizador, e não poderá ser explicada numa secção deste tipo. Como nos referiu que adquiriu o MSX para melhorar os seus conhecimentos de informática, pensamos que se procurar adaptar os referidos programas com a ajuda do manual do MSX e de um livro sobre o GW-BASIC, ou mesmo do que se disser na revista acerca do programa que pretenda adaptar, só terá a ganhar em conhecimentos e experiência no domínio da programação/utilização de computadores.

COMPRO/VENDO/TROCO

COMPRO

Compro computador gama Sinclair ZX, +, 2+ — Impressora A4 Amstrad/Sinclair ZX Spectrum. Telf. 324634/2180812.

Compro programa Art stúdio para Amstrad CPC6128 pode ser cópia ou original contactar Rui Manuel, Rua Rainha Santa Nº15 Armação de Pêra.

Compro qualquer modelo Amstrad PC. Telf. 9812317, a qualquer hora, Carlos Miguel Sousa.

Amstrad CPC464 Policromático a prestação. Contactar: António M. da Silva. Rua 1º de Maio LT.94 Parc.4 Fernão Ferro 2840 Seixal.

Disco rígido de 5, 10, 20 ou mais Mbytes + Placa controladora. contactar Paulo Nunes, Telf. 647333 9h-17h, Telf. 2075482, 20h-23h.

Amstrad PC ou PPC, 2ª mão, bom estado, 2 drives, mono, até 150 000$00. Telef. 9833728 Loures.

PC Amstrad 2 Drives. José Rodrigues. Casal do Rei 2300 Tomar.

VENDO

Vendo Spectra-vídeo HC 728, com 32k ROM e 80K RAM, 17 jogos-MSX. Preço 41 000$00. José Monteiro. R. Eduardo Espinheira. Tel. 9715560 Porto

Vendo programas originais de gestão e outro Software aplicado a pequenas e médias empresas — compatíveis — Tel. 031-54471 — F. Ribeiro.

MSX VG8010+Monit. poli VS0080+Gra. D6450 tudo Philips, bom estado e preço. Um só dono. Fernando J B Azevedo (052) 53098 — Depois 20H.

Spectrum 48K teclado Dk Tronics FDD 3000 Currah diskettes, cassetes pela melhor oferta. Telf. 08922688 José Alberto Portimão.

Executo programas para todos os fins. Cobol, Pascal ou Basic para compatíveis. Miguel Beirão 02-480857.

Computador Einstein, computador Philips p 3800 70MB diskettes 3.5". Telef. 531580. Av. 5 de Outubro 61, 3ºDto. 1000 Lisboa.

Vendo Amstrad PC1512 MDD com bastante software e impressora Citizen 120D. Tudo por 200 contos. Dá-se explicação Inic. Telef. 890784 Lisboa.

Amstrad CPC464 + Software. Contactar Lúcio Botelho. Telef. 617427.

PCW 8256 + Locoscript + Programa de contabilidade e outros programas de gestão. Telef. 497451 Porto — Paulo.

Jogos, cassete p/CPC 464 os mais recentes, bartos e c/ qualidade das melhores marcas de jogos. Luís Miguel Costa. Telef. (051) 22346.

Computador amstrad PC 1512 DD policromático + impressora DMP 3000 + software + manuais. Contactar Telef. (043) 53088 Santarém.

Executo todo o tipo PRO. PAS, COB, BAS, DB-Preçários, Relatórios, etc., em prog. adequados. Contactar Vasco Incio. Tr Tronco 95 4ºE Matos. Telef. 904562.

Spectrum Plus + Interface 1+2 microdrives + monitor Timex. Tudo 42 500$00. Telef. 9833728 Loures.

FDD3000 + Impressora 80 Colunas Timex + manuais e software como novo por 100 000$00. Tratar com Luís F. Bastos apartado 45 8700 Olhão.

Raquete ténis Dunlop 200-G (grafite). 1 mês uso. Urgente. Paulo Agostinho, Telef. 4103106. Facilidades de pagamento.

Amstrad PCW 8256 C/10 meses garantia. José M. M. Rodrigues, casal do Rei 2300 Tomar.

Executo programas originais à medida da sua necessidade para PC e compatíveis. Contactar pelo Telef. 20406 rede Santarém.

Fazemos programas para Amstrad PC compatíveis, Sócio & Sócio Software, Rua da Liberdade de 98 A, Telef. 419077, Guimarães.

Seikosha Gp-505 bom estado Telef. 2420536 depois das 19H.

Timex C 2048+ gravador Crown Tr-1 + Joystick Gunshot (oferta de 18 jogos) tudo como novo e só 30 000 escudos . Contactar 2047387 (01).

Polaroid 635 c/flach e célula foto-eléctrica p/controle autom. de luz. 6 000$00. Trata: Tavares — R. Manuel C. Ramalho, 16 — 2.º D Cartaxo.

Computador 48K Plus + Tv preto e branco 51 Cm + joystick + interfacer + 60 jogos. 50 contos, grátis gravador Sanyo no valor 6 contos. Telfs: 778177/660633

CPC 6128 + POC + Mini Office e programas tudo 80 contos ou troco por motorizada. Telef. 84197 (032). Viseu. Urgente.

TROCO

ITroco qualquer tipo de software para gama dos PC's compatíveis. João Pedro Ribeiro Santos, Rua Egas Moniz 3650 V. Nova Paiva, Telf. 54352.

Troco jogos e utilitários recentes para o CPC 664 Cass. ou disco. Francisco de Sousa. Av. Afonso Henriques 17 1ºDto. 2870 Montijo.

Troco jogos Amstrad CPC 464 em cassete. Contactar Marco por Carta. Papelaria Celinha, Lagoa 8400 Algarve.

Troco jogos e programas Amstrad CPC464. Contactar Marco. Papelaria Celinha, Lagoa 8400 Algarve.

Troco software Amstrad CPC464, em cassetes, contactar Marco por carta.
Papelaria Celinha, Lagoa 8400 Algarve.

Troco software para o PC1512. Contactar para: Rua Alexandre Herculano 4520 Feira ou pelo telf. 7648853. Nuno Moutinho.

Inforângelo Troca de software para compatíveis Amstrad, contacte Ap. 117 3080 Figueira da Foz.

Desejo trocar software p/ PC's compatíveis. Contacte Jorge Carvalho, Calçada dos Mestres 96 1ºDto 1000 Lisboa.

PCW — programas p/salários — IVA — Ficheiros de pessoal — Agenda ou específicos por encomenda. Troca/Cede/Estuda/Informa. Telef. 688351.

Troco jogos para o spectrum 48K procuro especialmente de treinar equipas de futebol.
Av. D. Dinis Bloco A 1ºDto 5 000 Vila Real.

PCW 8256 C10 Meses garantia por PC 2 drives. José Rodrigues. Casal do Rei 2300 tomar.

Software para IBM compatíveis. Contactar Maximilian Xavier. Telef. (01) 632411.

Mais de 30 jogos para o Amstrad CPC 464 contactar Lúcio Ferrão, Lote 201 11ºA Portela 2685 Sacavém ou Telef. 9436035 Lisboa

CPC 464. Vêm de França 400 jogos e utilitários. Carlos Pinto da Silva, Vale Pinheiro, Almas de Freire, Santa Clara 3 000 Coimbra.

Cassetes CPC 464 e Trucs, Rogéiro Alves V.
O Casulo Secarias — Silveira 2560 Torres Vedras.

Troco todo o tipo de Software para PC's. Contactar Pedro Oliveira. Alto Barreiros — 42 Santa Clara 3 000 Coimbra.
Telef. 039-812973.

— ... por outro lado, parece-me um pouco lento...

A ALTERNATIVA LÓGICA

Com o ATARI ST Você é o protagonista.
O microcomputador apenas uma valiosa ferramenta de trabalho.
Com a melhor relação custo/benefício pomos à sua disposição:

— um design inovador;
— a tecnologia mais avançada;
— potente software, incluindo o ambiente GEM, gerido por um «rato» de alta precisão.

E agora, pela primeira vez, através dos emuladores MS-DOS e MacIntosh, Você pode ainda aceder às duas maiores e melhores bibliotecas de aplicação existentes no mercado.
A preços que certamente não imaginaria.

520ST FM-512Kb RAM
Incluindo disquete 360Kb. . . a menos de 80 c.
C/ monitor monocromático
640 × 400 . . . . a menos de 120 c.
I.V.A. não incluído

MS-DOS e MacIntosh são marcas registadas da Microsoft Corporation e Apple Computer, Inc., respectivamente.

TIMING/ALLIANCE

TRÊS COMPUTADORES
NUM SÓ

DIVISÃO DE GRANDE DIFUSÃO
Av. Brasil, 147-A e B — 1700 LISBOA
Telef. 80 95 22 — Telex 64798 CEBITE — Fax 80 99 80
PORTO (02) 69 53 91